

# Mesh-free free-form lensing – I. Methodology and application to mass reconstruction

Julian Merten<sup>1,2,3★</sup>

<sup>1</sup>*Department of Physics, University of Oxford, Keble Road, Oxford OX1 3RH, UK*

<sup>2</sup>*Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109, USA*

<sup>3</sup>*California Institute of Technology, MC 249-17, Pasadena, CA 91125, USA*

Accepted 2016 June 10. Received 2016 June 10; in original form 2014 December 19

## ABSTRACT

Many applications and algorithms in the field of gravitational lensing make use of meshes with a finite number of nodes to analyse and manipulate data. Specific examples in lensing are astronomical CCD images in general, the reconstruction of density distributions from lensing data, lens–source plane mapping or the characterization and interpolation of a point spread function. We present a numerical framework to interpolate and differentiate in the mesh-free domain, defined by nodes with coordinates that follow no regular pattern. The framework is based on radial basis functions (RBFs) to smoothly represent data around the nodes. We demonstrate the performance of Gaussian RBF-based, mesh-free interpolation and differentiation, which reaches the sub-percent level in both cases. We use our newly developed framework to translate ideas of free-form mass reconstruction from lensing on to the mesh-free domain. By reconstructing a simulated mock lens we find that strong-lensing only reconstructions achieve <10 per cent accuracy in the areas where these constraints are available but provide poorer results when departing from these regions. Weak-lensing only reconstructions give <10 per cent accuracy outside the strong-lensing regime, but cannot resolve the inner core structure of the lens. Once both regimes are combined, accurate reconstructions can be achieved over the full field of view. The reconstruction of a simulated lens, using constraints that mimics real observations, yields accurate results in terms of surface-mass density, Navarro-Frenk-White profile (NFW) parameters, Einstein radius and magnification map recovery, encouraging the application of this method to real data.

**Key words:** gravitational lensing: strong – gravitational lensing: weak – methods: numerical – galaxies: clusters: general – dark matter – large-scale structure of Universe.

## 1 INTRODUCTION

Many techniques of astrophysical data analysis, although they in principle work on smooth and continuous data, are confined to a discrete numerical domain, which evaluates the input data using a finite number of analysis nodes. This numerical domain is usually referred to as a mesh and the coordinates of its nodes can have different dimensionality, depending on the application. In many cases the structure of these node coordinates follows a regular pattern, so a constant separation of nodes in each dimension. This regular pattern is convenient because it reduces the numerical complexity of the problem and simplifies many numerical algorithms since the spatial structure of the data is highly symmetric and easy to implement. However, many applications need a more sophisticated description of the spatial distribution of input data. A more general mesh layout

is provided by adaptive mesh refinement which increases the resolution of the mesh wherever this is allowed by the quality of the data and affordable in terms of CPU-time. The distances between nodes in each dimension is now adaptive but still follows a regular pattern. However, real astrophysical data, e.g. the distribution of galaxies or stars in a certain patch of the sky can be distributed in a very irregular fashion with density fluctuations, clusters and voids. When translating the input data on to a regular, structured mesh, this can lead to highly oversampled and partly unconstrained meshes or interpolation is needed, introducing the associated interpolation errors. The other extreme are undersampled meshes where averaging techniques compress the data on to a regular mesh. This comes at the price of smoothing out information and thus not making use of the full potential of the data.

In this work we present a framework which can deal with a mesh-free structure in the input data distribution. This means that the coordinates of the nodes follow no regular pattern and can have any values within the numerical domain. Two important types of data

\* E-mail: [julian.merten@physics.ox.ac.uk](mailto:julian.merten@physics.ox.ac.uk)

manipulation on such structures are interpolation and differentiation and we will introduce efficient algorithms which can achieve both of these tasks in any spatial dimension. The key to such a framework are radial basis functions (RBFs), functions which only depend on the distance of their evaluation points to certain reference points.

This work, which focuses on mass reconstruction from gravitational lensing, is only the first in a series, which will exploit our mesh-free numerical techniques. Two regimes are typically distinguished in lensing mass reconstruction. Strong lensing is usually confined to the inner-most core of the gravitational lens and produces spectacular observational constraints such as multiple images of the same source, gravitational arcs or even rings. The domain of weak lensing is further away from the centre of the lens but spans large areas and manifests itself by the weak distortion in the shape of background galaxies behind the lens. Reconstruction techniques are divided into two classes, although this distinction is by no means unique or even consistent in some cases. Parametric techniques (e.g. Kneib et al. 1996; Broadhurst et al. 2005; Smith et al. 2005; Halkola, Seitz & Pannella 2006; Jullo et al. 2007; Zitrin et al. 2009; Oguri 2010; Newman et al. 2013; Jullo et al. 2014; Monna et al. 2014; Johnson et al. 2014, for some recent examples) assume a parametric form of the underlying mass density distribution for the lens and typically make the assumption that light traces mass in the positioning of these parametric forms. On the other hand, free-form<sup>1</sup> methods (see e.g. Broadhurst, Taylor & Peacock 1995; Bartelmann et al. 1996; Abdelsalam, Saha & Williams 1998; Bridle et al. 1998; Seitz, Schneider & Bartelmann 1998; Bradač et al. 2005a; Cacciato et al. 2006; Liesenborgs, De Rijcke & Dejonghe 2006; Diego et al. 2007; Jee et al. 2007; Coe et al. 2008; Bradač et al. 2009; Merten et al. 2009; Williams & Saha 2011; Merten et al. 2011, 2015; Diego et al. 2015, for some recent examples) usually do not make this assumption and purely rely on the input data either based on weak lensing, strong lensing or a combination of the two. This is possible while using a reconstruction mesh and directly inverting the underlying equations describing lensing on this mesh. In the following, we introduce a free-form method combining weak and strong lensing, which uses our new mesh-free numerical framework. This method translates original ideas by Bartelmann et al. (1996), Seitz et al. (1998), Bradač et al. (2005a), Cacciato et al. (2006) and Merten et al. (2009) into the flexible and efficient mesh-free numerical domain. Alternative implementations of such ideas can e.g. be found in Bradač et al. (2009).

This work is structured as follows: in Section 2 we introduce RBFs and how they can be used to numerically interpolate and differentiate. In Section 3 we use the developed techniques to implement a free-form reconstruction algorithm that can be used in the mesh-free domain and consistently combines the regimes of weak and strong gravitational lensing. We test our implementation with numerically simulated data in Section 4 and we conclude in Section 5. In Appendix A we provide more details on the performance of interpolation with RBFs to the interested reader, as we do in Appendix B for mesh-free numerical differentiation. Appendix C gives some missing but not crucial details on the concrete implementation of the reconstruction algorithm. In many graphical illustrations in this paper we have to visualize mesh-free data. We do so by using the Voronoi tessellation of the evaluation points and by assigning a function value to each Voronoi cell, which refers to the function value at the respective coordinate. Throughout this work, we as-

<sup>1</sup> Sometimes also dubbed as non-parametric, which is misleading. In fact every mesh node is usually a free parameter rendering these methods highly parametric.

**Table 1.** A few common choices for radial basis functions.

Name	Functional form $\phi(r)$
Gaussian	$e^{-(\epsilon r)^2}$
Multiquadric	$\sqrt{1 + (\epsilon r)^2}$
Inverse multiquadric	$1/\sqrt{1 + (\epsilon r)^2}$
Inverse quadric	$1/(1 + (\epsilon r)^2)$
Polyharmonic spline	$r^{2m-1}$ $r^{2m} \log r$ with $m \in \mathbb{N}$

sume a flat  $\Lambda$ CDM cosmology with  $h = 0.7$ ,  $\Omega_\Lambda = 0.7$  and  $\Omega_m = 0.3$ . In Section 4.2, we refer to physical quantities of a simulated lens at  $z = 0.25$  where one arc second corresponds to 3.91 kpc.

## 2 RADIAL BASIS FUNCTIONS

In this general methodology section we will deal with functions which are defined on a finite number of evaluation points. Based on this set of points, we will interpolate functions in their numerical domain and calculate their derivatives. The general idea which enables us to do so is based on RBFs, where we expand the discretely defined functions into a set of radially dependent functions around the evaluation points. For a thorough discussion of the concept of RBFs see Fasshauer (2007) and references therein.

### 2.1 Unstructured meshes and mesh-free data

We define a mesh  $\mathcal{M}$  as a finite collection of  $N$  support points  $\mathbf{x}$

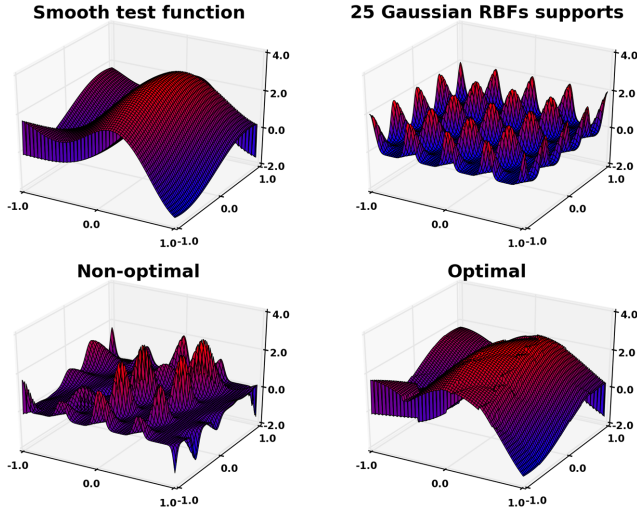
$$\mathcal{M} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]. \quad (1)$$

The dimensionality of the mesh is given by the number of coordinates  $n$  needed to define each support point  $\mathbf{x} = (x_1, x_2, \dots, x_d)$ . In most cases we will focus on the 2D case with  $\mathbf{x} = (x_1, x_2)$ , but the methodology presented in this section generalizes to any dimension.

This definition of a mesh  $\mathcal{M}$  is general in the sense that it includes cases with regularly distributed nodes or randomly distributed nodes. In this work we will show examples for both kinds of meshes, but generally focus on unstructured, meaning irregularly shaped mesh configurations. The range of coordinates depends on the numerical domain the mesh is defined on but for simplicity we will mostly restrict ourselves to coordinates  $x_i \in [-1, \dots, 1]$  with  $i$  depending on the dimensionality of the problem. Unstructured meshes can be still restricted when it comes to the distribution of their nodes. In the field of finite elements, for example, specific elements need to be formed with restrictions on the aspect ratios of their edges. For our purposes we do not have such restrictions, which is why we distinguish the notion of unstructured meshes from our, more general, case where the distribution of node coordinates follows no restrictions and which we call mesh-free.

### 2.2 Interpolation with RBFs

We analyse a function  $f$  defined at  $n$  nodes  $\mathbf{x}_1, \dots, \mathbf{x}_n$  and denote the values of the function at these points  $f(\mathbf{x}_1), \dots, f(\mathbf{x}_n)$  with  $f_1, \dots, f_n$ . We use RBFs  $\phi(\mathbf{x}) = \phi(\|\mathbf{x} - \mathbf{x}_0\|)$  to interpolate  $f$  to any position  $\mathbf{x}$  in the numerical domain. Throughout this work, radial distances are defined as the Euclidean norm  $L_2$  with the short-hand notation  $\|\mathbf{x} - \mathbf{x}_0\| = r$  for a given reference point  $\mathbf{x}_0$ . Typical choices for RBFs are listed in Table 1, but in this work we will restrict ourselves to Gaussian RBFs. In this case, the only free



**Figure 1.** A graphical illustration of expanding a 2D function into a set of 25 Gaussian RBFs. The top left panel shows the function to be interpolated defined on a unit mesh. The top right panel shows 25 Gaussian RBFs with an arbitrary amplitude of 2, an arbitrary shape parameter of 10 and placed on regularly spaced evaluation points. In the bottom left panel the amplitude of the Gaussians was set to the actual value of the test function at the position of the evaluation points, but the shape parameter was not optimized. In the bottom right panel, the amplitudes of the Gaussians were chosen by solving equation (3) and the shape parameter was optimized for this interpolation problem.

parameter,  $\epsilon$ , is called the shape parameter and has to be chosen carefully as will be discussed in great detail in Section 2.4 and Appendix A. For a more thorough discussion of the underlying mathematical concepts we refer to Fornberg, Larsson & Flyer (2011), Larsson et al. (2013), Fornberg, Lehto & Powell (2013), Flyer, Wright & Fornberg (2014) and references therein.

We write the interpolant  $\tilde{f}$  of the function  $f$  defined at nodes  $\mathbf{x}_1, \dots, \mathbf{x}_n$  as a weighted sum over RBFs

$$\tilde{f}(\mathbf{x}) = \sum_{i=1}^n \lambda_i \phi(\|\mathbf{x} - \mathbf{x}_i\|) \quad (2)$$

with weighting coefficients  $\lambda_i$ . Because of  $\tilde{f}(\mathbf{x}_i) = f_i$  we can calculate the coefficients  $\lambda$  by solving the linear system of equations

$$\underbrace{\begin{pmatrix} \phi(\|\mathbf{x}_1 - \mathbf{x}_1\|) & \cdots & \phi(\|\mathbf{x}_1 - \mathbf{x}_n\|) \\ \vdots & \ddots & \vdots \\ \phi(\|\mathbf{x}_n - \mathbf{x}_1\|) & \cdots & \phi(\|\mathbf{x}_n - \mathbf{x}_n\|) \end{pmatrix}}_{\mathcal{F}} \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{pmatrix} = \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix}. \quad (3)$$

It is important to note that for distinct nodes, this linear system cannot become singular, no matter how the nodes are scattered in any number of dimensions. Fig. 1 shows a graphical illustration of the expansion of a function into Gaussian RBFs around evaluation points. The accuracy of the interpolation is certainly dependent on the shape parameter and on the choice and number of nodes. We discuss the shape parameter further in Section 2.4 and present a detailed performance analysis of the RBF-based interpolation scheme in Appendix A.

### 2.3 Mesh-free numerical derivatives

Numerical derivatives are usually calculated by means of finite differencing (FD)

$$Df(\mathbf{x}) \approx \sum_{i=1}^n w_i f(\mathbf{x}_i) \quad (4)$$

for a linear differential operator  $D$  and with evaluation points  $\mathbf{x}_i$  defining a FD stencil. There are several ways of finding the weights  $w$  for the FD on regular meshes, ranging from the Lagrange interpolation polynomial, Taylor expansion, monomial test functions to the elegant Padé-algorithm. For a thorough description of all different techniques and the original references, see Fornberg (1998). Here we focus on monomial test functions since it will set the stage for mesh-free RBF-based FD later.

The motivation for the use of monomial test functions is to enforce that equation (4) holds exactly when  $f$  is a polynomial of degree  $n-1$ . In the 1D case, the weights in equation (4) are then given by the solution of the linear system of equations

$$\begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{n-1} & x_2^{n-1} & \cdots & x_n^{n-1} \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{pmatrix} = \begin{pmatrix} D1(\mathbf{x}) \\ Dx(\mathbf{x}) \\ \vdots \\ Dx^{n-1}(\mathbf{x}) \end{pmatrix} \quad (5)$$

where  $x_1, \dots, x_n$  are the evaluation points of the FD stencil and the differential operator  $D$  is applied to the monomial test functions  $x^{n-1}$  at the point of interest  $\mathbf{x}$ . This approach easily generalizes to higher dimensions by inserting also the other coordinate components of  $\mathbf{x}$  and mixed-component monomial test functions. The case of  $D = \Delta$  and  $n = 5$  recovers the well-known FD stencil on a regular mesh with node separation  $h$ , evaluated at  $\mathbf{x} = \mathbf{x}_0$

$$h^{-2} \begin{pmatrix} 1 & 1 & -4 & 1 & 1 \end{pmatrix} \begin{pmatrix} f(x_0 + h, y_0) \\ f(x_0 - h, y_0) \\ f(x_0) \\ f(x_0, y_0 + h) \\ f(x_0, y_0 - h) \end{pmatrix} \approx \Delta f(\mathbf{x}_0). \quad (6)$$

Inspired by this approach, we substitute the monomial test functions with RBFs and advance to a mesh-free formulation by centring the RBFs on the evaluation points of the function. In complete analogy to equation (5), this Ansatz leads to the following linear system of equations to find the FD weights

$$\mathcal{F} \begin{pmatrix} w_1 \\ \vdots \\ w_n \end{pmatrix} = \begin{pmatrix} D\phi(\|\mathbf{x} - \mathbf{x}_1\|) \\ \vdots \\ D\phi(\|\mathbf{x} - \mathbf{x}_n\|) \end{pmatrix}. \quad (7)$$

Monomial terms can be included again in order to increase the accuracy of the numerical derivatives. In the 2D case with monomial terms up to second order the FD weights are given by the solution

to

$$\begin{pmatrix}
 1 & x_1 & y_1 & x_1^2 & y_1^2 & x_1 y_1 \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
 1 & x_n & y_n & x_n^2 & y_n^2 & x_n y_n \\
 1 \cdots 1 \\
 x_1 \cdots x_n \\
 y_1 \cdots y_n \\
 x_1^2 \cdots x_n^2 & 0 \\
 y_1^2 \cdots y_n^2 \\
 x_1 y_1 \cdots y_n y_n
 \end{pmatrix}
 \begin{pmatrix}
 w_1 \\
 \vdots \\
 w_n \\
 w_{n+1} \\
 w_{n+2} \\
 w_{n+3} \\
 w_{n+4} \\
 w_{n+5} \\
 w_{n+6}
 \end{pmatrix}
 =
 \begin{pmatrix}
 D\phi(\|x - x_1\|) \\
 \vdots \\
 D\phi(\|x - x_n\|) \\
 D1(x) \\
 Dx(x) \\
 Dy(x) \\
 Dx^2(x) \\
 Dy^2(x) \\
 D(xy)(x)
 \end{pmatrix} \quad (8)$$

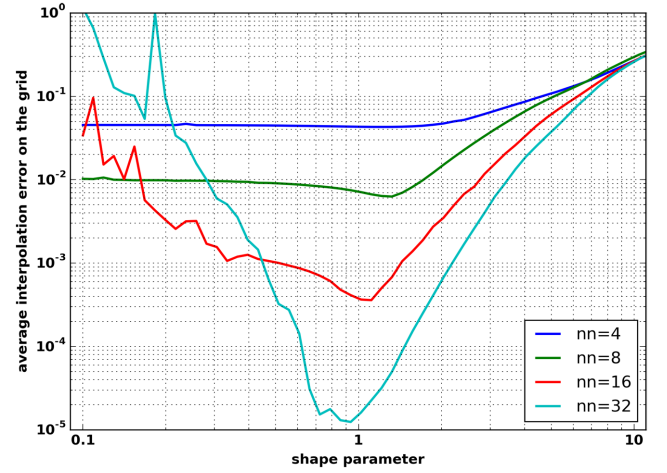
Only  $w_1, \dots, w_n$  are used as weights in equation (4). The remaining values  $w_{n+1}, \dots, w_{n+6}$  have no obvious meaning. This 2D RBF FD scheme with up to second-order monomial terms applies to most of our applications in this paper and is the scheme that we use in this work from now on, if not otherwise stated. The performance of RBF-derived FD stencils is discussed in the next section and in great detail in Appendix B.

## 2.4 Discussion of the shape parameter

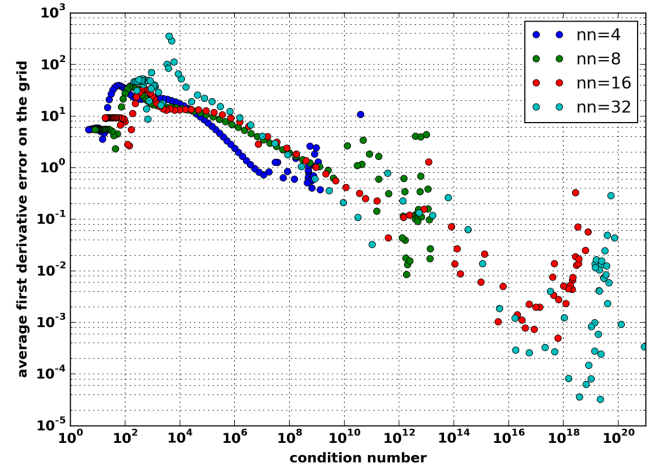
The shape parameter controls the width of the Gaussian RBF and crucially controls the accuracy of the RBF application no matter if one uses RBFs for interpolation or FD. For a complete discussion of the shape parameter we refer the interested reader to the work by Fornberg et al. (2011), Fornberg et al. (2013) and Larsson et al. (2013). In the following we will heuristically analyse the effect of the shape parameter by introducing the test function

$$f(x, y) = 1 + \sin(4x) + \cos(3x) + \sin(2y). \quad (9)$$

Based on 180 randomly chosen evaluation points, we interpolate the test function to 900 random locations (nodes) on the unit disc. For a more complete description of the test setup we refer the interested reader to Appendix A. In Fig. 2 we show the shape parameter dependence of the average and maximum relative error of the interpolation and also include the dependence on the number of nearest neighbour evaluation points used to carry out the interpolation for each interpolant evaluation point. As one can clearly see, there is an optimal choice for the shape parameter in order to achieve maximum accuracy. This choice depends on the actual node coordinates, and the number of evaluation points. Once the shape parameter is optimized (Fornberg et al. 2013), very high accuracies in the interpolation can be achieved, reaching an average relative interpolation error  $\sim 10^{-5}$ . We discuss this in more detail in Appendix A, with another test function that is specifically chosen for our purposes. As a final remark on the theory of RBF interpolation we point out, that the approach can be further optimized by choosing a spatially



**Figure 2.** The accuracy of the test function interpolation as a function of the shape parameter and of the number of nearest neighbours used to perform the interpolation. The interpolation mesh consists of 900 nodes and of 180 support points. The interpolation function is shown in equation (9).



**Figure 3.** The accuracy of the first numerical  $x$  derivative of the test function (equation 9) as a function of the condition number and the number of nearest neighbours in the FD stencil.

varying shape parameter. Since the choice of a variable shape parameter, depending on the distribution of evaluation points, is not trivial (Fornberg & Zuev 2007) we focus for now on the special case of a single  $\epsilon$  value. We will present a more general method, that adaptively varies the shape parameter, in the course of the development of our mesh-free methods.

For RBF-based FD stencils, it has been shown that the respective FD weights only give accurate results if the condition number  $C^2$  of the coefficient matrix in equation (7) is close to critical,<sup>3</sup> and thus depends on the machine precision of the implementation (Fornberg et al. 2013). One has to carefully monitor this condition number, which can be directly controlled by the shape parameter. In Fig. 3 we show the accuracy of the numerical first  $x$  derivative as a

<sup>2</sup> The condition number of the coefficient matrix is defined as the ratio between the largest and smallest singular value of the SVD decomposition of the coefficient matrix.

<sup>3</sup> This means that  $\log(C) \gtrsim \epsilon$ , where  $\epsilon$  is the machine precision of your numerical implementation.



function of this condition number and the number of nearest neighbours in the stencil. The accuracy is rapidly increasing when approaching the ill-condition point, where  $C$  is approaching the critical value, of the system. Once the system is ill-conditioned, or close to that, the accuracy is decreasing again and shows irregular behaviour. This is discussed in much more detail in Fornberg et al. (2013) and we give a more detailed heuristic assessment in Appendix B. It is also worth noting that the condition number is independent of the differential operator  $D$ , as can be seen from equation (7). In the following we mostly use 16 to 32 nearest neighbours to define the FD stencil in each evaluation point and therefore we adjust the shape parameter to keep the condition number of  $\mathcal{F}$  in the  $10^{16}$ – $10^{17}$  region. In order to avoid the dependence on shape parameter altogether, the use of polyharmonic spline (PHS) type RBFs (compare Table 1) is a sensible approach. The performance of PHS-RBFs in RBF-FD applications is currently under investigation and shows promising results (Bengt Fornberg, private communication). We will implement these RBFs in the numerical framework that we present in the next section and will perform a series of tests to analyse the accuracy and feasibility of PHS-type RBFs for our purposes.

## 2.5 Numerical implementation

We briefly describe our own numerical implementation<sup>4</sup> of mesh-free interpolation and differentiation with RBFs. The library is written in C++ and mainly provides, among several helper routines, two classes. The central class describes a collection of  $n$  nodes with arbitrary coordinates in either 1D, 2D or 3D and is initially defined by an input vector  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]$ . For both, interpolation and differentiation in this mesh-free domain, it is important to find the nearest neighbours of each node  $\mathbf{x}_i$ . The mesh-free nodes class provides this functionality by means of a kdd-tree based ordering algorithm. In our implementation we use the publicly available library FLANN<sup>5</sup> (Muja & Lowe 2009). The mesh-free nodes class then provides useful features like node coordinate queries and returns node index vectors for the nearest neighbours of each node.

The second central class implements RBFs. The shape parameter, the origin of the coordinate system and the dimensionality of the problem (1D, 2D or 3D) can be set. The user can evaluate the RBF and its derivatives at different coordinates. Currently, only Gaussian RBFs are implemented but the class uses inheritance from virtual functions to generally represent an RBF and allows the user to implement more general cases by providing the functional form of the RBF and its derivatives.

The combination of the mesh-free nodes and the RBF class enables the functionality, which was described in the course of this section. Once the functional form of an RBF is provided, mesh-free interpolation to arbitrary evaluation points is enabled and the FD weights for each evaluation point can be calculated and returned as a matrix  $\mathcal{W}_{lk}$ . This allows the user to differentiate a function  $\Psi_k = [\Psi(\mathbf{x}_1), \Psi(\mathbf{x}_2), \dots, \Psi(\mathbf{x}_n)]$  by a simple matrix multiplication

$$D\Psi_l \approx \mathcal{W}_{lk} \Psi_k. \quad (10)$$

Currently all differential operators  $D$  up to third order are implemented in our classes.

## 3 LENSING MASS RECONSTRUCTION

We apply the RBF framework to a concrete astrophysical application, mass reconstruction from gravitational lensing. After a short lensing primer we show how different constraints from gravitational lensing can be combined in a free-form way by using a mesh-free approach.

### 3.1 Lensing primer

Einstein's theory of general relativity predicts the deflection of light rays due to gravitational potentials (see e.g. Bartelmann 2010, for a complete derivation). By introducing the thin-lens approximation, which assumes that the distances between objects in the lensing scenario are much larger than the spatial extent of these objects, the lens mapping can be described by a lens equation

$$\boldsymbol{\beta} = \boldsymbol{\theta} - \boldsymbol{\alpha}(\boldsymbol{\theta}). \quad (11)$$

This central equation describes how the 2D angular position in the source plane  $\boldsymbol{\beta} = (\beta_1, \beta_2)$  is mapped by a deflection angle  $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$  on to the angular coordinates  $\boldsymbol{\theta} = (\theta_1, \theta_2)$  in the lens plane. The deflection angle can be related to a lensing potential

$$\psi(\boldsymbol{\theta}) := \frac{1}{\pi} \int d^2\theta' \frac{\Sigma(D_0\boldsymbol{\theta})}{\Sigma_{cr}} \ln|\boldsymbol{\theta} - \boldsymbol{\theta}'|, \quad (12)$$

that inherits the surface-mass density of the lens  $\Sigma(D_0\boldsymbol{\theta})$ . The cosmological background model enters this equation through the critical surface mass density for lensing given by

$$\Sigma_{cr} = \frac{c^2}{4\pi G} \frac{D_s}{D_l D_{ls}}, \quad (13)$$

where  $c$  is the speed of light and  $G$  is Newton's constant. The angular diameter distance between observer and lens  $D_l$ , between observer and source  $D_s$ , and between lens and source  $D_{ls}$  set the geometry of the lensing scenario.

When introducing the edth operators (Newman & Penrose 1962)  $\partial := (\frac{\partial}{\partial\theta_1} + i\frac{\partial}{\partial\theta_2})$  and  $\partial^* := (\frac{\partial}{\partial\theta_1} - i\frac{\partial}{\partial\theta_2})$ , lensing quantities are easily related to the lensing potential (see e.g. Bartelmann & Schneider 2001, and references therein)

$$\begin{aligned} \alpha &:= \partial\psi & s &= 1 \\ 2\gamma &:= \partial\partial\psi & s &= 2 \\ 2\kappa &:= \partial\partial^*\psi & s &= 0 \end{aligned} \quad (14)$$

where  $\alpha$  is the deflection angle,  $\gamma$  is called the complex shear and the scalar quantity  $\kappa$  is called convergence. The spin-parameter  $s$  describes the transformation properties of each quantity under rotations of the coordinate frame.

The regime of weak gravitational lensing is governed by small distortions in the shape of background galaxies, which are observationally measured as complex ellipticities  $\epsilon(\boldsymbol{\theta})$ . Once we introduce the reduced shear

$$g := \frac{\gamma}{1 - \kappa}, \quad (15)$$

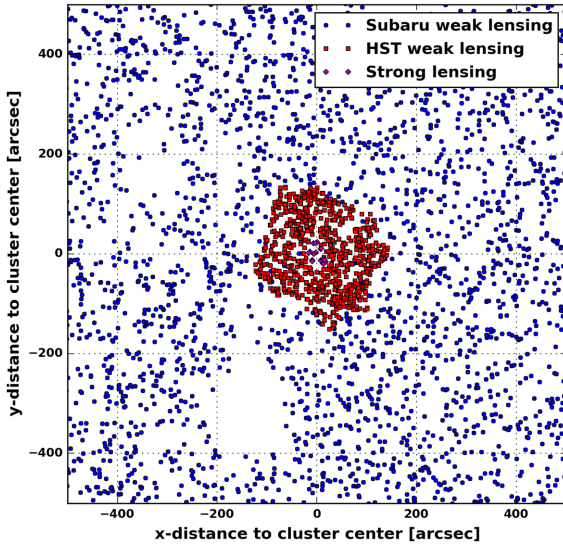
we can establish the connection between localized ellipticity averages over an ensemble of sources and the properties of the lens

$$\langle\epsilon\rangle = \begin{cases} g & \text{for } |g| \leq 1 \\ \frac{1}{g^*} & \text{for } |g| > 1. \end{cases} \quad (16)$$

The local averages are necessary to separate the lensing signal from random orientations due to the intrinsic ellipticity of galaxies. For a thorough review of weak lensing and its applications we refer to

<sup>4</sup> <https://bitbucket.org/jmerten82/libmfree>

<sup>5</sup> <http://www.cs.ubc.ca/research/flann/>



**Figure 4.** The distribution of lensing constraints in the galaxy cluster Abell 383 as seen by CLASH (Zitrin et al. 2011; Postman et al. 2012; Merten et al. 2015).

Bartelmann & Schneider (2001, and references therein) and for a discussion of systematic effects in weak lensing studies to Kitching et al. (2012), Massey et al. (2013) and Mandelbaum et al. (2014).

In the strong-lensing regime the lens equation becomes nonlinear, multiple images of the same source can form and shape distortions are not small any more. This leads, in some cases, to the formation of spectacular gravitational arcs or even rings in the vicinity of a strong lens. The spatial extent of this regime, close to the core of the lens where densities are highest, is indicated by the critical curve at a given redshift. It is defined by the roots of the determinant of the lensing Jacobian

$$\det \mathcal{A} = (1 - \kappa)^2 - \gamma^* \gamma. \quad (17)$$

### 3.2 Combining lensing constraints in the mesh-free domain

The toolkit developed in Section 2 can be used to perform mesh-free lensing reconstructions. That is to recover the underlying total mass distribution of a lens from weak- and strong-lensing constraints. Equations (14), (15) and (17) show that all these effects can be related to spatial derivatives of the lensing potential. Since we can calculate numerical derivatives from randomly distributed nodes, we construct a reconstruction method by assigning each lensing observable an evaluation point in the mesh-free domain. A typical distribution of such observables is shown in Fig. 4 with the galaxy cluster Abell 383, as seen by the Cluster Lensing and Supernova Survey with Hubble (CLASH) (Zitrin et al. 2011; Postman et al. 2012; Merten et al. 2015), as an example.

Once we have chosen the number of nearest neighbours in the FD stencil, we can express equation (4) as a matrix operation (equation 10). In the specific case of the lensing quantities related to the discretized lensing potential  $\psi = [\psi(\mathbf{x}_1), \psi(\mathbf{x}_2), \dots, \psi(\mathbf{x}_N)]$  we find

$$\alpha_l^{1,2} = \mathcal{D}_{lk}^{1,2} \psi_k \quad (18)$$

$$\gamma_l^{1,2} = \mathcal{G}_{lk}^{1,2} \psi_k \quad (19)$$

$$\kappa_l = \mathcal{K}_{lk} \psi_k \quad (20)$$

where  $\mathcal{D}$ ,  $\mathcal{G}$  and  $\mathcal{K}$  contain the FD weights for the differential operators in equation (14). In order to recover the lensing potential from the input constraints we define a  $\chi^2$ -function, which relates the lensing observations at each evaluation point to the lensing potential. We will define all components of this function in different lensing regimes later on. In order to find the lensing potential which is most likely to have caused the observed lensing effects we minimize the  $\chi^2$ -function with respect to the potential values at each evaluation point

$$\frac{\partial \chi^2(\psi_k)}{\partial \psi_l} \stackrel{!}{=} 0; \quad (21)$$

where we use

$$\frac{\partial \mathcal{W}_{lk} \psi_k}{\partial \psi_l} = \delta_{lk} \quad (22)$$

for any matrix representation  $\mathcal{W}$  of a FD stencil.

#### 3.2.1 Weak lensing

It is our goal to combine several lensing constraints into a joint reconstruction algorithm and we start with the weak-lensing contribution. Equation (15) shows that average measured ellipticities of background galaxies are directly related to the reduced shear of the lens. Hence, we write the weak-lensing term as

$$\chi_w^2 = \sum_{i,j} ((\epsilon) - g(\psi))_i \mathcal{C}_{ij}^{-1} ((\epsilon) - g(\psi))_j, \quad (23)$$

where the indices  $i, j$  run over all weak-lensing evaluation points. It has been shown in Bradač et al. (2005a) and Merten et al. (2009) how the minimization of such a  $\chi^2$  with respect to the discretized lensing potential can be written as a linear system of equations, while using equations (19), (20) and (22). We refer the interested reader to appendix A of Merten et al. (2009), which shows the full derivation but we give additional information in Appendix C. We have to mention that we limited ourselves to the case  $|g| \leq 1$ . Reliable weak-lensing shape measurements near the very centre of strong lenses are challenging and may introduce unwanted systematic effects. Henceforth, we exclude the regime where  $|g| > 1$  from our weak-lensing analysis, but rely on constraints from strong lensing in these areas. In principle, however, and given a reliable shape measurement in this regime, weak-lensing reconstruction nodes with  $|g| > 1$  can be included in the reconstruction scheme, as e.g. shown in Bradač et al. (2005a).

The covariance matrix of the weak-lensing data  $\mathcal{C}_{ij}$  deserves special attention. It is well-known that galaxies carry an intrinsic ellipticity with a standard deviation of  $\sigma_\epsilon \sim 0.3$  (e.g. Chang et al. 2013), which is not induced by lensing. One way of incorporating this into the  $\chi^2$  minimization is to assume that the galaxy intrinsic ellipticities are uncorrelated, which results in a diagonal covariance matrix with the canonical value of  $0.3^2$  for all its non-zero elements. However, in the presence of noise, this approach leads to a very poor recovery of the lensing potential, as we will show later on. A different approach exploits the fact that the intrinsic ellipticity of galaxies has no preferred orientation.<sup>6</sup> To a given weak-lensing evaluation point we do not assign the ellipticity value of a single galaxy but we perform a distance-weighted average over an ensemble of nearest neighbours with respect to the point of interest.

<sup>6</sup> This statement of course ignores the effect of intrinsic alignments of galaxies in wide-field shear surveys (e.g. Hirata & Seljak 2004), but for our purposes of cluster lensing by individual objects, it is certainly justified.

By doing so, the undirected, intrinsic ellipticity signal averages out and the coherent lensing signal remains. This procedure obviously introduces correlations between the neighbouring pixels which were used to define the ellipticity samples. We calculate this sample covariance following equation (15) of Merten et al. (2009) and take it into account by summing over the full covariance in the  $\chi^2$ -minimization.

### 3.2.2 Strong lensing

One of the biggest advantages and indeed the biggest motivation for a mesh-free reconstruction algorithm is the fact that the distribution of evaluation points is intrinsically adaptive. This is important because different lensing constraints are confined to quite different length scales, as is clearly seen in Fig. 4. The mesh-free approach allows us to place evaluation points where data is available. In the case of weak lensing, this spans the entire field of the lens, with usually increased resolution towards the centre when high-quality data from e.g. the *Hubble Space Telescope* (HST) is available. Strong lensing is confined to the very core of the lens and allows for a very finely grained recovery of the lensing potential if many strong-lensing features are observable.

One constraint related to strong lensing are multiple-image systems. Our  $\chi^2$ -minimization term is similar to Bradač et al. (2005a) but differs in some details. The general idea is based on the fact that different images  $i$  of the same multiple-image system should be mapped back to the same position in the source plane. Therefore, we write a  $\chi^2$ -term

$$\chi_m^2 = \sum_{i=1}^{N_s} \left( \frac{\beta_i(\psi) - \langle \beta \rangle(\psi)}{\sigma_{i,m}} \right)^2 \quad (24)$$

where  $\beta_i$  is the source-plane position of each image of the system and

$$\langle \beta \rangle = \frac{1}{N_s} \sum_{i=1}^{N_s} \beta_i \quad (25)$$

is the average source-plane position of all images in the system. The total number of images in a single multiple-image system is  $N_s$ . In the last two equations we can use equation (11) to replace the source-plane position  $\beta$  with the observed lens-plane position  $\theta$  and the deflection angle  $\alpha$ , which carries the wanted dependence on the lensing potential

$$\chi_m^2 = \sum_{i=1}^{N_s} \frac{1}{\sigma_s^2} \left( \theta_i - \alpha_i(\psi) - \frac{1}{N_s} \sum_{j=1}^{N_s} (\theta_j - \alpha_j(\psi)) \right)^2. \quad (26)$$

Here,  $\sigma_s$  is the tolerated positional error on each image position in the source plane. The result of minimizing this  $\chi^2$ -function with respect to the lensing potential is shown in Appendix C. In the presence of more than one multiple-image system, one adds a  $\chi^2$ -term for each system, respectively.

The second strong-lensing constraint are estimates on the position of the critical curve of the lens. This has been discussed in detail in e.g. (Jullo et al. 2007; Merten et al. 2009, 2011, 2015). The corresponding  $\chi^2$ -term enforces the lensing Jacobian to vanish for pixels which are assigned to be part of the critical curve and which are indicated in the following by a pixel index  $c$

$$\chi_s^2(\psi) = \sum_{c=1}^{N_c} \frac{|\det \mathcal{A}(\psi)|_c^2}{\sigma_{c,s}^2}. \quad (27)$$

The total number of these estimators is  $N_c$  and the error  $\sigma_s^2$  derives from a positional error that is assigned to the critical curve estimator. We approximate it via

$$\sigma_s \approx \left. \frac{\partial \det \mathcal{A}}{\partial \theta} \right|_{\theta_E} \delta \theta \approx \frac{\delta \theta}{\theta_E}, \quad (28)$$

where  $\theta_E$  is an estimate for the Einstein radius of the lens. The minimization of the  $\chi^2$ -function related to this constraint can also be related to a linear system of equations using equations (19), (20) and (22) and we again leave the actual calculation to appendix A of Merten et al. (2009). When using critical line estimators as a constraint, one has to keep in mind that they are not a direct observable in a lensing scenario. The advantages and caveats when using these constraints, together with the accuracy of lensing reconstructions has been discussed earlier (e.g. Cacciato et al. 2006; Merten et al. 2009; Meneghetti et al. 2010; Merten et al. 2015). We still include this constraint in this paper since it is a useful feature in a lensing reconstruction method. One can for example incorporate constraints from another pure strong-lensing method into the current reconstruction via these estimators.

### 3.2.3 Implementation

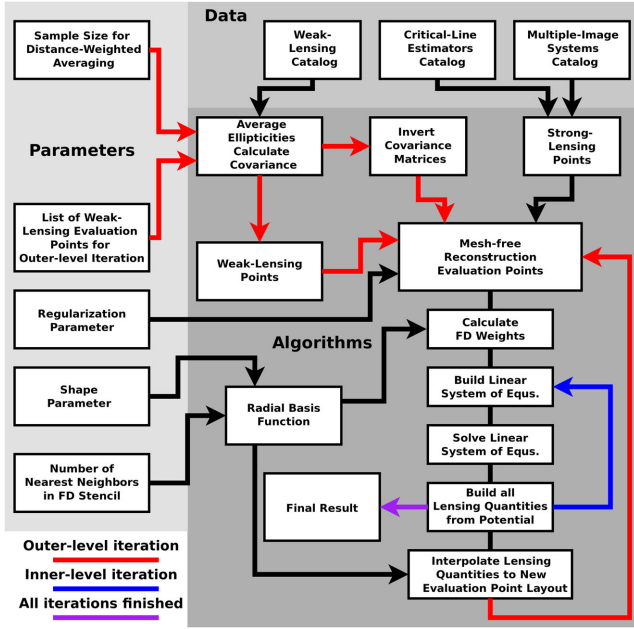
In order to find the lensing potential which causes the joint observations of all weak- and strong-lensing terms we sum over the independent contributions of the  $\chi^2$  function and find a single linear system of equations. The solution to this linear system is the mesh-free representation of the reconstructed lensing potential, from which all other quantities of interest can be derived using equations (18)–(20).

In order to guarantee a smooth reconstruction in the presence of noisy weak-lensing data we introduce an outer-level iteration, following the scheme of Bradač et al. (2005a), which is also used in Merten et al. (2009). We define a regularization term in the  $\chi^2$ -function which controls the reconstruction in such a way that the result will not diverge strongly from a well-defined regularization condition. In our case, this condition is set by pre-defined convergence  $\kappa_{\text{reg}}$  and shear  $\gamma_{\text{reg}}$  solutions

$$\chi_{c,\text{reg}}^2 = \sum_{i=1}^N \eta_i^c (\kappa_i(\psi) - \kappa_{\text{reg}})^2 \quad (29)$$

$$\chi_{s,\text{reg}}^2 = \sum_{i=1}^N \eta_i^s (\gamma_i(\psi) - \gamma_{\text{reg}})^2. \quad (30)$$

In the summation above,  $i$  runs over all evaluation points and it should be noted that in this implementation the strength of the regularization  $\eta$  can be set for each evaluation point individually and can be set to different values for convergence and shear regularization, respectively. The idea of the outer-level iteration is then to start with only few weak-lensing evaluation points and to average ellipticities of a large sample of weak-lensing sources for each of these nodes. This results in a coarse but almost shape-noise free reconstruction. In the following steps, the number of nodes is continuously increased, resulting in smaller ellipticity samples but relying on a reconstruction that is convergence- and shear-regularized on the result of the former reconstruction step. This outer-level iteration effectively reduces the noise-level in the reconstruction, as shown in Bradač et al. (2005a) and Merten et al. (2009), and as we will explore in our accuracy tests later on.



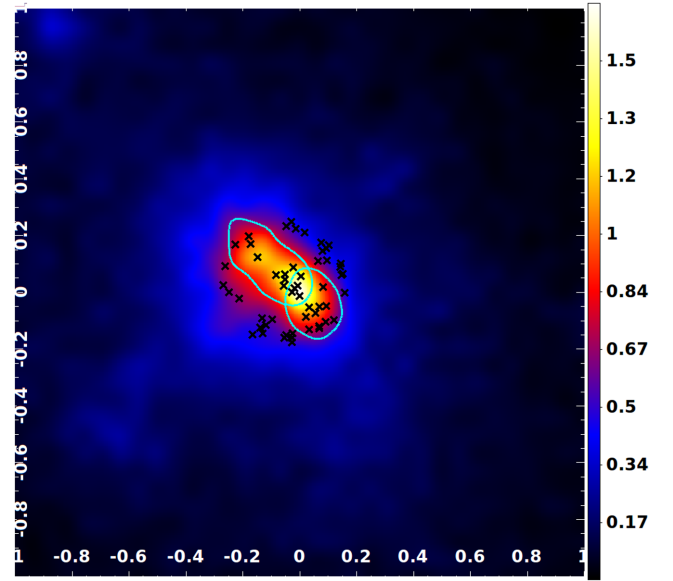
**Figure 5.** A flowchart of the reconstruction process using weak- and strong-lensing constraints.

Ultimately, we are solving the linear system of equations which is calculated from the minimization of the  $\chi^2$ -function

$$\chi^2(\psi) = \chi_w^2(\psi) + \chi_s^2(\psi) + \chi_m^2(\psi) + \chi_{s,reg}^2(\psi) + \chi_{c,reg}^2. \quad (31)$$

One notes that the weak lensing and the critical-line estimator term contain nonlinear contributions. We account for this by introducing an inner-level iteration, following the scheme of Schneider & Seitz (1995). During each inner-level reconstruction iteration, nonlinear terms in the  $\chi^2$ -function are isolated and held constant in order to solve the linear system of equations. New estimates for convergence and shear are calculated from this solution and new approximations for the nonlinear terms are inserted as constants into the linear system of equations. This iteration converges after two to five reconstruction steps.

A complete flowchart of the reconstruction algorithm is shown in Fig. 5. Initially, the weak-lensing catalogue is read and depending on the stage of the outer-level iteration, ellipticity values are averaged to define the weak-lensing evaluation points. These are then combined with the strong-lensing evaluation points, which directly derive from the critical-line estimator and multiple-image system catalogues. For all these nodes, the FD stencils are calculated using RBFs and the reconstruction is performed, including the inner-level iteration. This procedure is repeated with increasing resolution until all outer-level iteration steps are completed. Each reconstructed result is interpolated to the next larger mesh-free outer-level configuration using RBFs. The convergence and shear maps derived from this interpolation serve as the regularization template for the next step. The very first regularization template depends on the reconstruction and the field of view of the data but, in most cases, a flat and zero convergence and shear template suffices. However, a more sophisticated choice for the initial prior is also possible resulting in more complicated initial convergence and shear regularization templates.



**Figure 6.** The convergence map of our mock lens on the unit square. The cyan line in the centre of the image is the critical line of the lens for a fiducial redshift of  $z_s = 1.0$ . The black crosses indicate the positions of multiple images that we use to reconstruct the lens.

## 4 ACCURACY TESTS WITH MOCK LENSES

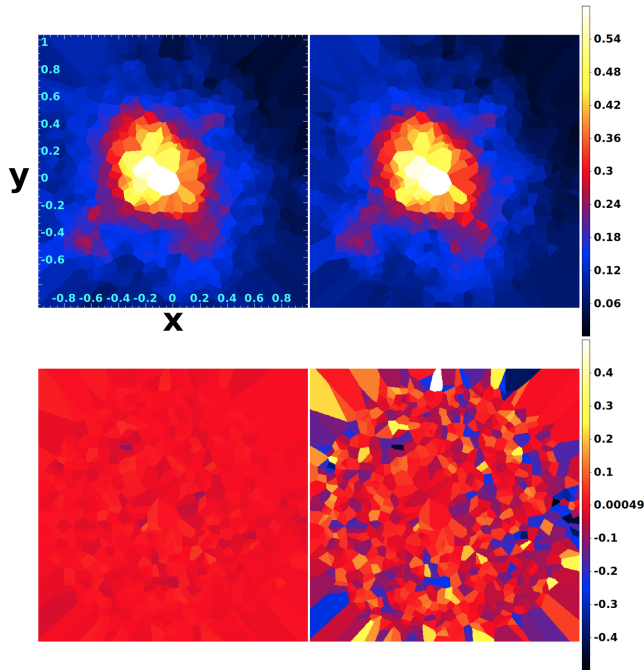
We use two different test scenarios to check the implementation and performance of our reconstruction method. A simple toy model lens provides the framework to check the basic implementation in various stages. A simple toy model based test is followed by a much more realistic lensing scenario, which is based on a full ray-tracing simulation of a simulated cluster-sized halo. There we also mimic several observational effects and sources of noise.

### 4.1 Toy model lens

We use a numerically simulated, cluster-sized lens to provide a simple proof of concept for the applicability of our theoretical concept. This mock lens is described in more detail in Bartelmann et al. (1998) and was already used for accuracy tests in Cacciato et al. (2006) and Merten et al. (2009). The surface-mass density map of this lens is shown in Fig. 6. The side length of the field of view is  $5 \text{ Mpc h}^{-1}$  or 18 arcmin at the lens' redshift of  $z = 0.35$ . The Einstein radius of the lens is  $\theta_E \sim 30 \text{ arcsec}$  for a source redshift of  $z_s = 1.0$ . In the following, especially in the figures of this section, we will scale these distances to dimensionless coordinates by mapping them into the unit-square with side length 2. From the known deflection angle fields, we sample the following lensing catalogues to serve as input for our test reconstructions:

- (i) 9000 complex shear values at random positions in the field. This refers to a background-galaxy density of  $\sim 25 \text{ arcmin}^{-1}$ .
- (ii) The same catalogue of 9000 weak-lensing shear values but with an added shape-noise component. This noise is sampled randomly in each shear component from a Gaussian distribution with a standard deviation of 0.2 (compare Section 3.2).
- (iii) 55 multiple images, by randomly placing 5 point sources inside the inner caustic and 10 point sources in between the inner and the outer caustic of the lens.





**Figure 7.** The result of the weak-lensing only reconstruction of the mock lens as a Voronoi representation. This reconstruction is performed without any noise contribution in the weak-lensing input data. The left-hand panel shows the convergence map of the reconstruction. The right-hand panel shows the relative error in the reconstructed convergence when compared to the true input at the evaluation points of the reconstruction.

(iv) 20 critical line estimators by randomly sampling points in the field for which the determinant of the lensing Jacobian (equation 17) vanishes within the limits of the numerical precision.

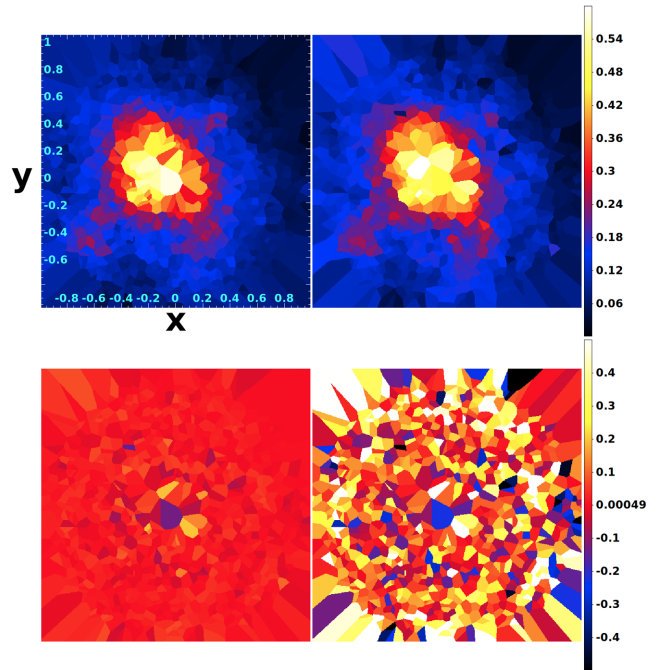
All these lensing constraints are placed at a fiducial redshift of  $z_s = 1.0$ .

In the following we will investigate how well we can reconstruct the underlying mass distribution from weak lensing alone, where we investigate the influence of important complications to such analyses.

As a word of caution we want to stress that these tests can only be seen as a proof of concept. More thorough and realistic tests with fully realistic lensing scenarios (e.g. Meneghetti et al. 2010) have to follow, together with a full comparison to other mass reconstruction methods. Furthermore, the readiness of the method to be applied to real astronomical data has to be shown in such a follow-up analysis.

#### 4.1.1 Weak-lensing tests

We perform a pure weak-lensing reconstruction of the mock lens using the shear catalogue without any shape noise contribution first. From the 9000 ideal shear values we randomly pick 900 to serve as evaluation points of the mesh-free reconstruction. Since the data contains no noise component, no outer-level iteration is needed, but we still assign the fiducial value  $\sigma_\epsilon = 0.3$  to each weak-lensing constraint. However, we use the reduced shear as weak-lensing input which demands the inner-level iteration to compensate for the nonlinear contributions to the  $\chi^2$ -minimization. In order to correct for the mass-sheet degeneracy we force the very upper-right corner of the reconstructed region to approach a convergence value of zero. We present the result in Fig. 7 where the top panel shows the real convergence map of the mock lens on the left and the reconstructed



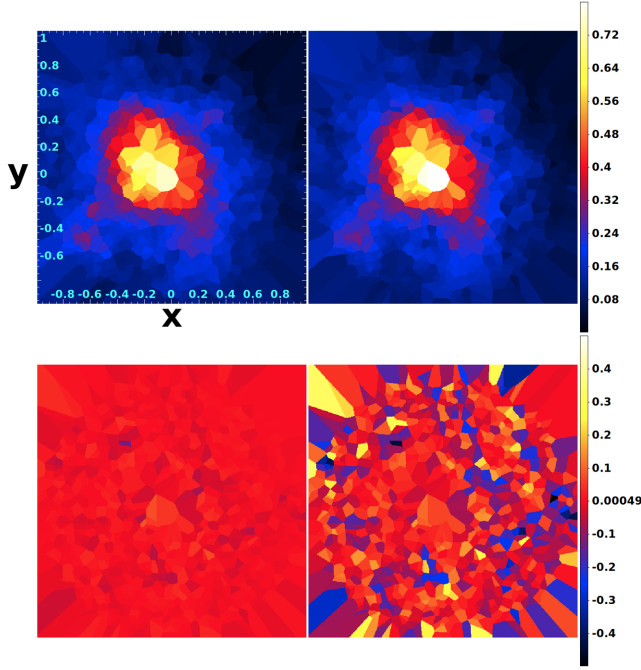
**Figure 8.** The result of the weak-lensing only reconstruction of the mock lens as a Voronoi representation. The difference to Fig. 7 is the realistic redshift distribution of the sources in the weak-lensing input catalogue. The reconstruction assumes the sources to sit at a single redshift, which introduces a significant bias.

convergence map on the right. Both maps follow the same resolution based on the 900 weak-lensing evaluation points. The bottom panels show the absolute difference between the reconstructed convergence map and the real one on the left and the relative difference to the real map on the right. The general agreement is quite striking, as it should be under these perfect test conditions. The average absolute difference between the maps is  $-0.002$  and the average relative difference is  $-0.02$ . We performed this oversimplified reconstruction to serve as a benchmark for our additional reconstructions which include typical complications for weak-lensing analyses.

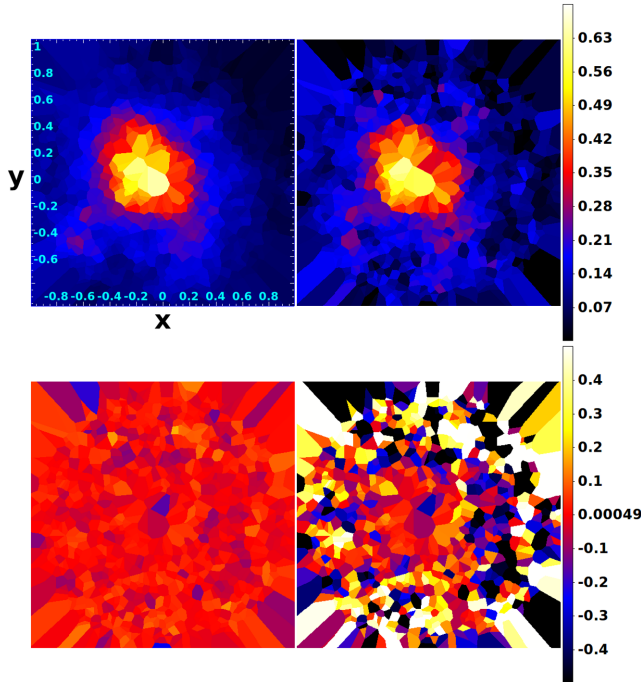
To make the scenario more realistic we now distribute the weak-lensing sources according to the photometric redshift distribution of background sources in a real galaxy cluster.<sup>7</sup> Once we reconstruct this input catalogue while wrongly assuming that all sources are still placed at a single redshift we find the result presented in Fig. 8. The wrong assumption introduces a significant source of error, pushing the average absolute difference in the maps to 0.01 and the average relative difference to 0.12. However, our method is able to deal with different redshifts for each reconstruction node by scaling each node to a common fiducial redshift (see e.g. equations 8 and 12 in Merten et al. (2009)). Once we incorporate the knowledge about the photometric redshifts we find the result shown in Fig. 9, which reduces the average, absolute difference in the maps back to  $-0.002$  and the average, relative error to  $-0.019$ . Please note that both, the real and the reconstructed convergence map are now scaled to a fiducial redshift of  $\infty$ .

We test for another important complication by using a reduced shear catalogue which contains shape noise. The result is shown in Fig. 10. The outer-level iteration scheme, which is described in Section 2.5, is implemented to minimize the effect of shape noise

<sup>7</sup> Abell 209 from the CLASH survey, compare Merten et al. (2015).



**Figure 9.** The result of the weak-lensing only reconstruction of the mock lens as a Voronoi representation. This reconstruction uses the same input catalogue as Fig. 8 but accounts for the redshift distribution in the reconstruction. The convergence map are now scaled to a fiducial redshift of  $\infty$  and the initial accuracy level seen in Fig. 7 is now restored.



**Figure 10.** The result of the weak-lensing only reconstruction of the mock lens as a Voronoi representation. This reconstruction is performed with a realistic noise contribution in the weak-lensing input data. The left-hand panel shows the convergence map of the reconstruction. The right-hand panel shows the relative error in the reconstructed convergence when compared to the true input at the evaluation points of the reconstruction.

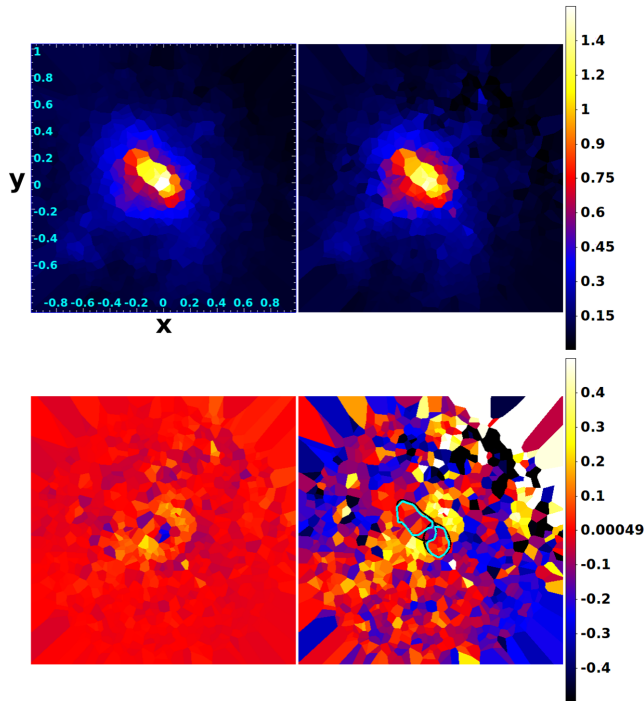
on the reconstruction. We define six different refinement levels by starting with 150 weak-lensing evaluation points and gradually add 150 more points until we reach the target resolution of 900 nodes. The weak-lensing catalogue for this reconstruction is the full ellipticity sample, containing 9000 measurements and shape noise. For each of these six outer-level iterations we perform a distant-weighted average over 80, 50, 30, 22, 18 and 15 weak-lensing shear values, respectively, in order to extract the lensing signal from the noisy data. For the first step with 150 reconstruction cells we regularize on a flat and zero convergence and shear field as initial step. Later on we regularize on the interpolated results of the former, more smooth reconstruction step to avoid overfitting. For each outer-level resolution we perform three inner-level iterations, which is enough for the reconstruction to converge. The average absolute error in the convergence is  $-0.005$  and the relative error drops to a value of  $-0.04$ . The maximum absolute and relative errors are  $-0.27$  and  $2.44$ , respectively, which are found in the outskirts of the field where the lensing signal is weakest and the shear values are largely dominated by shape noise.

#### 4.1.2 Weak- and strong-lensing tests

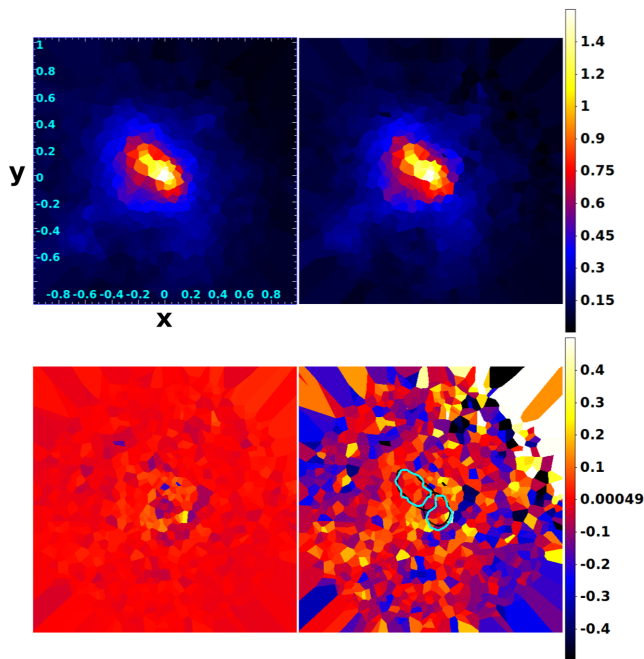
This last effort of the testing program brings the pieces together and performs combined weak- and strong-lensing reconstructions. We again draw 900 ellipticity measurements from the shape-noise free weak-lensing catalogue and also use the 55 multiple-image systems in a first joint reconstruction. The result of this reconstruction is shown in Fig. 11 and it is immediately obvious that, while the field size is identical to the weak-lensing only reconstruction, the resolution in the central area of the lens is increased drastically due to the additional evaluation points defined by the positions of the multiple images. The reconstruction yields an average absolute convergence error of  $-0.007$  and the average of the relative error is  $0.004$ . The maximum absolute and relative convergence errors are  $-0.29$  and  $-2.4$ , respectively. In the bottom right panel of Fig. 11 we also show the reconstructed critical curve of the lens and compare it with the real critical curve for our fiducial redshift of  $z_s = 1.0$ . Only small deviations between the two curves are present and the fact that the reconstructed critical curve gets split into two parts is only due to the limited number of evaluation points. In general, the accuracy of the reconstruction is at the 5–10 per cent level, with the clear exception of areas just outside the critical curve where the reconstruction overestimates the convergence by 15–20 per cent especially in the areas around the  $(x, y) = (0.0, -0.2)$  and the  $(0.2, 0.2)$  coordinate. We overcome this shortcoming by making use of the one feature in the reconstruction that we have not used, yet. We add the additional 20 sample points of the critical curve of the cluster. We show the reconstruction that adds these constraints in Fig. 12 and again find an excellent reconstruction but with much reduced inaccuracies around the critical curve of the cluster. The average absolute and relative error is now  $-0.005$  and  $-0.009$ , respectively. The maximum absolute error is  $-0.25$  and the maximum relative error is  $1.75$ . The recovery of the critical curve is still excellent as we also show in the bottom right panel of Fig. 12.

## 4.2 Realistic ray-tracing scenario

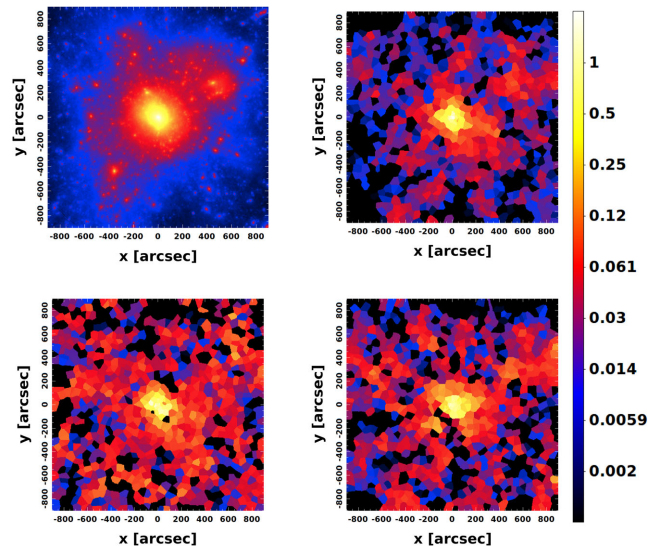
We now move to a much more realistic lensing scenario, created with the SKYLENS pipeline (Meneghetti et al. 2008). The lens in this ray-tracing approach is provided by one 2D deflection angle map of the cluster *g5699754\_G\_79235*, which is part of the simulated



**Figure 11.** The Voronoi representation of the joint weak- and strong lensing reconstruction using 900 reduced shear values and 55 multiple-image system. The top panels show the convergence of the real lens on the left and the reconstructed convergence map on the right. Both maps follow a resolution which is defined by the 955 input constraints. Shown in the bottom panels are differences between the real and reconstructed maps in terms of an absolute error on the left and a relative error with the real map as a reference on the right. Also shown in the bottom right panel is the reconstructed critical line in cyan and the real critical line for a source redshift of 1.0 in black. The black line can barely be seen since it is well overlaid by the reconstructed line.



**Figure 12.** This figure is identical to Fig. 11 but uses in the underlying reconstruction additional 20 constraints on the position of the critical line of the cluster.



**Figure 13.** The real and reconstructed realistic lensing scenario. The *top left* panel shows the real convergence map of the simulated lens for a source redshift of  $z_s = 2.0$ . The total field size is 0.5 deg and corresponds to a physical scale of 7.04 Mpc. The *top right* panel shows the convergence map of our fiducial reconstruction scaled to the same source redshift. The two *bottom* panels show two randomly chosen reconstruction realizations based on input catalogue resampling.

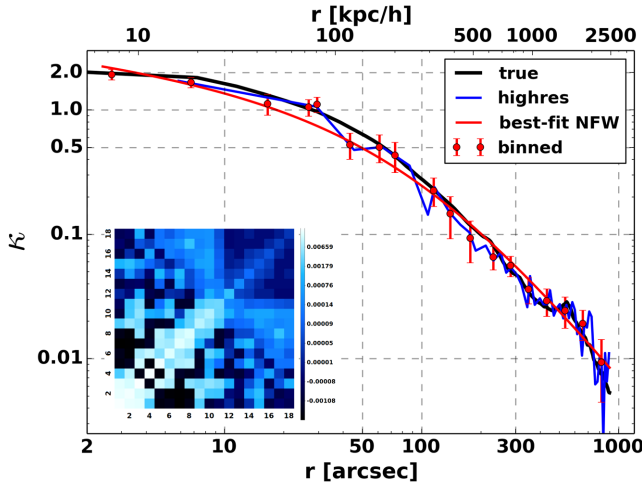
cluster suite described in Bonafede et al. (2011) and Fabjan et al. (2011). The simulated field of view is 0.5 deg on a side and the corresponding surface-mass density map along our chosen line of sight can be seen in Fig. 13. Following the example of Meneghetti et al. (2010) we create the following lensing constraints in the field:

- (i) 22752 galaxies with weak-lensing shape measurements, which corresponds to a galaxy density of  $\sim 25 \text{ arcmin}^{-1}$ . In contrast to the toy model case of Section 4.1, these galaxies are not randomly positioned any more, but their spatial distribution follows a realistic, cosmological clustering scheme and is affected by shifts due to lensing. More importantly, their intrinsic shape and redshift distribution follows real observations and their shape measurement is affected by real systematic effects (see Meneghetti et al. 2008, 2010, for details about the ray-tracing method).
- (ii) Ten multiple image systems, three of which produce five images, while the rest produces three. The redshifts of these systems cover the range from 0.970 to 3.636 in roughly equal steps.

#### 4.2.1 Reconstruction and error estimation

We apply the methodology outlined in Section 4.1 to the catalogues produced by the realistic lensing scenario. In order to deal with the effects of shape-noise in the weak-lensing catalogue and to avoid noise overfitting we start the reconstruction with 200 weak-lensing nodes, homogeneously covering the full field. The centre of each node is chosen to be the shape measurement in the vicinity with the largest shear inverse variance. Around all centres we perform a weighted average of the 100 nearest neighbours in the shear catalogue. In the strong-lensing regime we assign a reconstruction node to each of the 36 multiple images. We iteratively increase the resolution from 236 to 1036 reconstruction nodes in steps of 200, while in each step we decrease the number of averaged shears by a factor of 0.7. Each iteration keeps memory of the earlier step through regularization with a strength which is chosen to match the





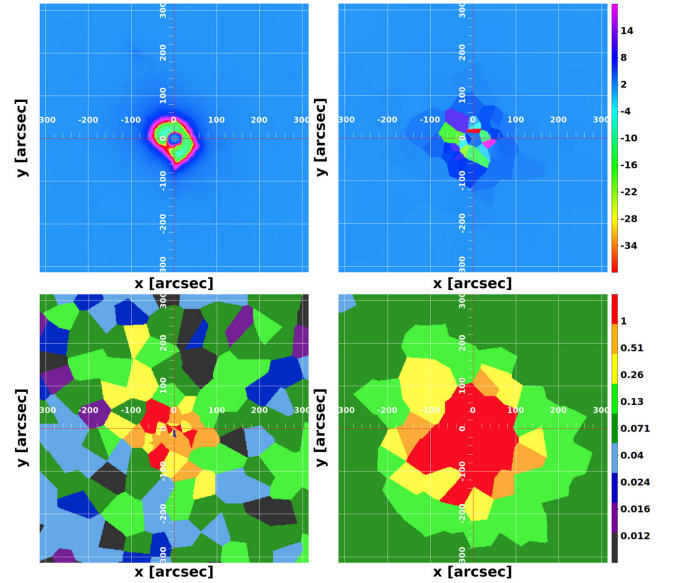
**Figure 14.** The convergence profile of the real cluster (black line) and of the reconstruction using two binning schemes for a source redshift of  $z_s = 2.0$ . The fine binning scheme (blue line) follows nicely even smaller substructure components but is quite noisy towards the outskirts. We use the logarithmic, coarser binning scheme (red points) to derive 2D NFW fitting parameters. The error bars are the square root of the diagonal elements in the bin-to-bin covariance matrix, which is shown in the bottom-left inset. The best-fitted NFW profile is overlaid as the red line.

weak-lensing weight set by covariance matrix in equation (23). In each of these outer level iterations we perform three inner level iterations. In the last outer level iteration step, our unstructured distribution of reconstruction nodes results in a minimum distance between neighbouring nodes of 1.1 arcsec and a maximum distance of 78.7 arcsec. The result of this reconstruction can be seen in Fig. 13, where we show the real convergence distribution of the simulated cluster next to our reconstruction. One can clearly see that the reconstruction suffers from the effect of shape noise but shows a qualitatively good agreement with the real matter distribution.

In order to assign realistic error bars to our reconstruction we resample our input catalogues (e.g. van Waerbeke 2000; Bradač et al. 2005b; Meneghetti et al. 2010; Merten et al. 2015). We create 250 reconstruction realizations by bootstrapping the weak-lensing input catalogue and by uniformly sampling a redshift range for each weak-lensing and strong-lensing constraint. We define the allowed range within 10 per cent below and above a constraint’s real redshift. We show two examples for the resampled realizations in Fig. 13. The total sample of 250 realizations builds the base to determine the errors in the following analyses. In the following we refer to the original, not resampled, reconstruction as the fiducial model. The total number of resamplings is mostly constrained by runtime considerations. However, we did confirm that the size of the relevant error bars is stable, when compared to smaller (e.g. 200 resamplings) sets of realizations.

#### 4.2.2 Mass profile and NFW parameter recovery

We derive surface-mass density profiles from both, the real projected matter distribution and our reconstruction. Shown in Fig. 14 are the profiles as a function of distance from the densest point in the simulation. We perform two different radial binning schemes for our reconstruction. One fine, linear binning which is able to follow all substructure variations in the reconstruction, but gets very noisy especially in the outskirts of the field where the signal



**Figure 15.** Real and reconstructed magnification maps in the *top left* and *top right* panel, respectively. All maps are scaled to a source redshift of  $z_s = 2.0$ . The *bottom left* panel shows the relative difference between the two maps above, evaluated at the positions of the reconstruction nodes. The *bottom right* panel shows the standard deviation of the relative difference, as evaluated from the 250 reconstruction realizations.

becomes weak. Hence, we also apply a coarser, logarithmically spaced binning scheme, which we will use later on for a parametric fit to the profile. The covariance matrix for the bins is derived from the profile analysis for all of our 250 resampled realizations. A first, visual, inspection shows that our reconstructed profile follows nicely within the error bars the true profile.

As a quantitative check we fit an NFW profile (Navarro, Frenk & White 1996, 1997) to the data. Since we are not interested in the potential biases that the assumption of spherical symmetry introduces when fitting 2D data with profiles derived from 3D simulations (see Meneghetti et al. 2014, and references therein for a discussion), we first derive the 2D NFW parameters for the true profile. We do so by assuming spherical symmetry while projecting the NFW profile and fitting it to the convergence profile. Following the commonly used parametrization of the NFW profile, this delivers a total mass  $M_{200} = 1.90 \times 10^{15} M_\odot$  and a concentration  $c_{200} = 5.46$  where both quantities are evaluated using a radius where the average density of the halo is 200 times the critical density of the Universe. By applying the same formalism to our reconstruction we find  $M_{200} = 1.83 \pm 0.15 \times 10^{15} M_\odot$  and  $c_{200} = 4.8 \pm 1.0$ . Both, the reconstructed mass and concentration are in good agreement with the true 2D values within their  $1\sigma$  error bars. The best-fitted, projected NFW profile is also overlaid in Fig. 14.

#### 4.2.3 Magnification map

For many applications including the exploration of the high-redshift Universe (e.g. Bradač et al. 2009; Zheng et al. 2012; Bradley et al. 2014) or the study of lensed Supernovae (e.g. Patel et al. 2014; Kelly et al. 2015; Rodney et al. 2015), it is very important to have magnification estimates for a lens at specific locations in the image plane. In Fig. 15, we show the real magnification map of the simulated halo as a zoom on the central region where the magnification starts deviating significantly from unity. The magnification values

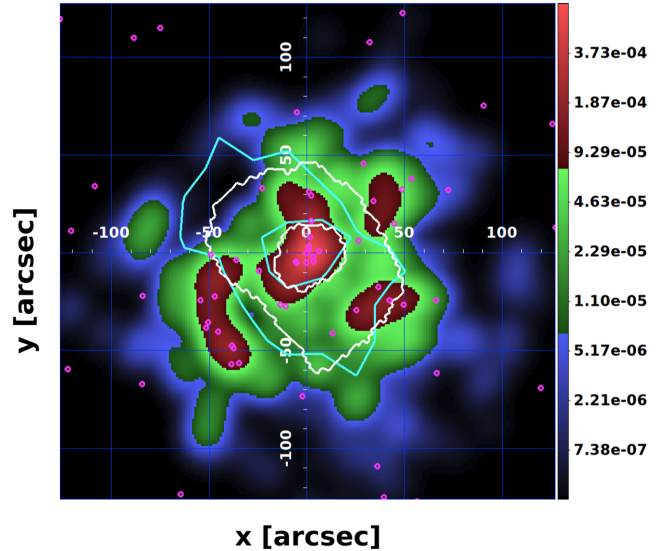


are derived for a fixed source redshift of  $z_s = 2.0$ . In the same figure we show the reconstructed magnification map for our fiducial reconstruction scaled to the same source redshift. For a quantitative comparison we show in the bottom panel of Fig. 15 the relative error when comparing the reconstructed magnification value to the real one. The general trend of this comparison can be described as follows. In the outskirts of the field the difference to the true value is well below 10 per cent and mostly below 5 per cent. When approaching the innermost 200 arcsec of the field, the error increases to 10–20 per cent and increases further to about 50 per cent near and within the critical line of the cluster, which is at  $\sim 50$  arcsec at this redshift. For some reconstruction nodes the relative error can be well above 100 per cent. These large discrepancies are expected close to the critical line of the cluster, where the lensing effects become extremely nonlinear. To explore the significance of these deviations we calculate the relative errors in magnification for all of our reconstruction realizations and calculate the scatter in each reconstruction node. The result is also shown in Fig. 15, indicating that all observed discrepancies between the reconstruction and the truth are well within the expected error bars.

#### 4.2.4 Critical curve and Einstein radius

As the last analysis for the realistic lensing scenario we look into the reconstruction of the critical curve, again for a source redshift of  $z_s = 2.0$ . First, we calculate the true Einstein radius of the halo by sampling its critical line with 1040 points, calculating the distance of each sample point and taking the sample average. Following this approach the Einstein radius is  $r_E = 46.5$  arcsec. For our fiducial reconstruction, we sequentially check for each pairing between a reconstruction node and its four nearest neighbours if the sign of the Jacobian determinant (equation 17) changes. If it does, we assign a point to the critical curve at the centre of the connecting line between the node and the respective neighbour. From this set of points we calculate the Einstein radius in the same way as described earlier. In order to assign an error bar, we repeat the procedure for all reconstruction realizations. The resulting Einstein radius of  $r_E = 47 \pm 4$  arcsec is in excellent agreement with the real value.

We show the full critical curve of the system in Fig. 16, together with the reconstruction from the fiducial model. The colour-coded background of the figure shows the probability distribution function for a point in the lens plane to be part of the critical curve. It is derived from a Gaussian kernel density estimator applied to the distribution of points assigned as part of the critical curve for all 250 reconstruction realizations. One can see that the areas of highest probability nicely follow the real position of the critical curve with two exceptions. The upper-left edge of the critical line, around coordinate  $(-50, 50)$  arcsec is not recovered at all and there is a clear misidentification of the critical line around the coordinate at  $(-50, -50)$  arcsec. We think that these two discrepancies can be explained with the distribution of the reconstruction nodes, which is also overlaid in the figure. The part of the critical curve which is recovered by the fiducial model, but seems unlikely given the error model clearly lies in an area of the field which is not well sampled by the reconstruction nodes. The opposite is the case where the contours of high probability clearly deviate from the real critical curve. Here the dense sampling of reconstruction nodes seems to steer some reconstruction realizations away from the true position. Although the general recovery of the critical curve, especially for the fiducial model, is good, this result motivates the exploration of



**Figure 16.** The critical curves of the halo for a source redshift of  $z_s = 2.0$ . The white line shows the true position, while the cyan line shows the reconstruction from our fiducial model. The colour of the background encodes the probability distribution function of a point in the lens plane to be part of the critical curve in the reconstruction. The pink circles show the distribution of reconstruction nodes near the critical curve of the halo.

alternative reconstruction node placements, which we reserve for a more detailed follow up work.

This finalizes our collection of realistic tests to determine the performance of the new, RBF-based methodology. We have shown that the method performs well in recovering the density profile, NFW parameters, the magnification map, the Einstein radius and with some caveats also the critical curve of the simulation. However, we want to point out that future studies are necessary beyond the general presentation of methodology in this work. Upcoming tests will include a much less massive lens, bigger variations in the number of multiple images and the effect of catastrophic redshift errors. Furthermore, we want to investigate alternative strategies of placing the reconstruction nodes, especially in the strong-lensing regime.

## 5 CONCLUSIONS

In this work we introduced a framework for the mesh-free interpolation and differentiation of functions. This framework is an important and potentially powerful tool for applications in the field of gravitational lensing, because input data does usually not follow any regular pattern in its spatial distribution and is confined to very different length scales. The particular examples for this problem in this work are the regimes of weak- and strong-gravitational lensing.

Our implementation of mesh-free interpolation and differentiation is based on the concept of RBFs. Specifically, we use Gaussian RBFs, although our methodology is not restricted to this one class of RBFs. We convincingly proved the performance of our implementation in Section 2 and Appendices A and B. We showed the importance of a well vetted shape parameter of the Gaussian RBF, depending on evaluation-points layout and application and showed that by using an increasing number of nodes in the nearest-neighbours stencils, higher accuracies can be achieved with the drawback of longer runtime. If all these parameters are chosen appropriately, the accuracy of our interpolation and differentiation routines is well below the percent level.

Using the new techniques to express mesh-free numerical derivatives we implemented a novel method for mass reconstruction from gravitational lensing. We translated the initial ideas of Bartelmann et al. (1996); Bradač et al. (2005a); Cacciato et al. (2006) and Merten et al. (2009) into the realm of a mesh-free and intrinsically adaptive reconstruction and tested the performance of this approach with a simple mock lens. In Section 4.1 we showed with a simple toy-model lens that we are able to reconstruct the lens based on 55 multiple images Weak-lensing constraints, when used as a stand-alone input, allow a reconstruction also at the 5–10 per cent accuracy level but do not cover the strong-lensing core of the lens. The main problem in weak-lensing analyses though is the presence of shape noise in the input data. We showed the performance of the well-established two-level iteration scheme of Bradač et al. (2005a) and Merten et al. (2009) in order to deal with noisy data in Fig. 10 and showed our ability to account for the redshift distribution of sources in Fig. 9. Finally, we combine weak- and strong-lensing constraints to achieve an accurate reconstruction of the mock lens over all relevant length scales. A much more relevant test in terms of applicability to real data was performed in Section 4.2. We showed that we can reproduce the mass of a simulated, massive halo with a precision of 8 per cent while using realistic weak- and strong-lensing input constraints. The same precision is achieved in the recovery of the Einstein radius. While we also recover the concentration, magnification map and the critical curve of the halo within our error bars, the results are less precise. Future studies will show how they will improve with a larger number of strong-lensing input constraints. An important test, for example in the context of the on-going HST Frontier Fields.<sup>8</sup>

An application to known, real observed lenses has to follow together with a comparison to other reconstruction techniques based on gravitational lensing. Furthermore, we are planning to apply our method not only to mass-reconstruction applications but extend our work to lens–source plane mapping. While lensing features are usually observed on very regular meshes in the lens plane, due to the pixelization scheme of CCD images, the lens mapping transforms this regular pattern into a very irregular one in the source plane. This is a problem both in ray-tracing simulations (Meneghetti et al. 2008) and in the source-plane reconstructions of lensed sources (see e.g. Dye & Warren 2005; Vegetti & Koopmans 2009; Tagore & Keeton 2014, and references therein) Finally, we will apply our reconstruction method to real data and we will explore the usefulness of our approach in the field of point spread function interpolation, which also deals with the irregular pattern of star positions in the fields of astronomical observations. Improvements to our general implementation may stem from the introduction of a spatially varying shape parameter (Fornberg & Zuev 2007), although most recent developments in applied mathematics may allow us to discard the shape parameter altogether when using PHS-type RBFs (Bengt Fornberg, private communication).

## ACKNOWLEDGEMENTS

I want to send a warm thank you to Bengt Fornberg for helping me to understand and implement the concept of radial basis functions for the purpose of finite differencing. I also thank Matthias Bartelmann, Massimo Meneghetti, and Leonidas Moustakas for inspiring discussions and Stefano Borgani for providing the numerical simulation used in our realistic ray-tracing scenario. This research was carried out at the Jet Propulsion Laboratory, Cali-

fornia Institute of Technology, under a contract with NASA and I acknowledge support from NASA Grants HST-GO-13343.05-A and HST-GO-13386.13-A. The research leading to these results has received funding from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme (FP7/2007-2013) under REA Grant agreement number 627288.

## REFERENCES

- Abdelsalam H. M., Saha P., Williams L. L. R., 1998, *MNRAS*, 294, 734  
 Bartelmann M., 2010, *Class. Quantum Gravity*, 27, 233001  
 Bartelmann M., Schneider P., 2001, *Phys. Rep.*, 340, 291  
 Bartelmann M., Narayan R., Seitz S., Schneider P., 1996, *ApJ*, 464, L115  
 Bartelmann M., Huss A., Colberg J. M., Jenkins A., Pearce F. R., 1998, *A&A*, 330, 1  
 Bonafede A., Dolag K., Stasyszyn F., Murante G., Borgani S., 2011, *MNRAS*, 418, 2234  
 Bradač M., Schneider P., Lombardi M., Erben T., 2005a, *A&A*, 437, 39  
 Bradač M. et al., 2005b, *A&A*, 437, 49  
 Bradač M. et al., 2009, *ApJ*, 706, 1201  
 Bradley L. D. et al., 2014, *ApJ*, 792, 76  
 Bridle S. L., Hobson M. P., Lasenby A. N., Saunders R., 1998, *MNRAS*, 299, 895  
 Broadhurst T. J., Taylor A. N., Peacock J. A., 1995, *ApJ*, 438, 49  
 Broadhurst T. et al., 2005, *ApJ*, 621, 53  
 Cacciato M., Bartelmann M., Meneghetti M., Moscardini L., 2006, *A&A*, 458, 349  
 Chang C. et al., 2013, *MNRAS*, 434, 2121  
 Coe D., Fuselier E., Benítez N., Broadhurst T., Frye B., Ford H., 2008, *ApJ*, 681, 814  
 Diego J. M., Tegmark M., Protopapas P., Sandvik H. B., 2007, *MNRAS*, 375, 958  
 Diego J. M. et al., 2015, *MNRAS*, 446, 683  
 Dye S., Warren S. J., 2005, *ApJ*, 623, 31  
 Fabjan D., Borgani S., Rasia E., Bonafede A., Dolag K., Murante G., Tornatore L., 2011, *MNRAS*, 416, 801  
 Fasshauer G. F., 2007, *Meshfree Approximation Methods with MATLAB*, World Scientific Publishing Company. Available at: <http://www.worldscientific.com/worldscibooks/10.1142/6437>  
 Flyer N., Wright G., Fornberg B., 2014, in Freedon W., Nashed M. Z., Sonar T., eds, *Handbook of Geomathematics*, Springer-Verlag, Berlin  
 Fornberg B., 1998, *SIAM Rev.*, 40, 685  
 Fornberg B., Zuev J., 2007, *Comput. Math. Appl.*, 54, 379  
 Fornberg B., Larsson E., Flyer N., 2011, *SIAM J. Sci. Comput.*, 33, 869  
 Fornberg B., Lehto E., Powell C., 2013, *Comput. Math. Appl.*, 65, 627  
 Halkola A., Seitz S., Pannella M., 2006, *MNRAS*, 372, 1425  
 Hirata C. M., Seljak U., 2004, *Phys. Rev. D*, 70, 063526  
 Jee M. J. et al., 2007, *ApJ*, 661, 728  
 Johnson T. L., Sharon K., Bayliss M. B., Gladders M. D., Coe D., Ebeling H., 2014, *ApJ*, 797, 48  
 Jullo E., Kneib J., Limousin M., Elíasdóttir Á., Marshall P. J., Verdugo T., 2007, *New J. Phys.*, 9, 447  
 Jullo E., Pires S., Jauzac M., Kneib J.-P., 2014, *MNRAS*, 437, 3969  
 Kelly P. L. et al., 2015, *Science*, 347, 1123  
 Kitching T. D. et al., 2012, *MNRAS*, 423, 3163  
 Kneib J., Ellis R. S., Smail I., Couch W. J., Sharples R. M., 1996, *ApJ*, 471, 643  
 Larsson E., Lehto E., Heryodono A., Fornberg B., 2013, *SIAM J. Sci. Comput.*, 35, A2096  
 Liesenborgs J., De Rijcke S., Dejonghe H., 2006, *MNRAS*, 367, 1209  
 Mandelbaum R. et al., 2014, *ApJS*, 212, 5  
 Massey R. et al., 2013, *MNRAS*, 429, 661  
 Meneghetti M., Bolzonella M., Bartelmann M., Moscardini L., Tormen G., 2000, *MNRAS*, 314, 338  
 Meneghetti M. et al., 2008, *A&A*, 482, 403  
 Meneghetti M., Rasia E., Merten J., Bellagamba F., Ettori S., Mazzotta P., Dolag K., Marri S., 2010, *A&A*, 514, A93

<sup>8</sup> <http://www.stsci.edu/hst/campaigns/frontier-fields/>

- Meneghetti M. et al., 2014, *ApJ*, 797, 34  
Merten J., Cacciato M., Meneghetti M., Mignone C., Bartelmann M., 2009, *A&A*, 500, 681  
Merten J. et al., 2011, *MNRAS*, 417, 333  
Merten J. et al., 2015, *ApJ*, 806, 4  
Monna A. et al., 2014, *MNRAS*, 438, 1417  
Muja M., Lowe D. G., 2009, in *Int. Conf. Comput. Vis. Theor. Appl. (VISSAPP'09)*, INSTICC Press, p. 331  
Navarro J. F., Frenk C. S., White S. D. M., 1996, *ApJ*, 462, 563  
Navarro J. F., Frenk C. S., White S. D. M., 1997, *ApJ*, 490, 493  
Newman E., Penrose R., 1962, *J. Math. Phys.*, 3, 566  
Newman A. B., Treu T., Ellis R. S., Sand D. J., Nipoti C., Richard J., Jullo E., 2013, *ApJ*, 765, 24  
Oguri M., 2010, *PASJ*, 62, 1017  
Patel B. et al., 2014, *ApJ*, 786, 9  
Postman M. et al., 2012, *ApJS*, 199, 25  
Rodney S. A. et al., 2015, *ApJ*, 811, 70  
Schneider P., Seitz C., 1995, *A&A*, 294, 411  
Seitz S., Schneider P., Bartelmann M., 1998, *A&A*, 337, 325  
Smith G. P., Kneib J.-P., Smail I., Mazzotta P., Ebeling H., Czoske O., 2005, *MNRAS*, 359, 417  
Tagore A. S., Keeton C. R., 2014, *MNRAS*, 445, 694  
van Waerbeke L., 2000, *MNRAS*, 313, 524  
Vegetti S., Koopmans L. V. E., 2009, *MNRAS*, 392, 945  
Williams L. L. R., Saha P., 2011, *MNRAS*, 415, 488  
Wright C. O., Brainerd T. G., 2000, *ApJ*, 534, 34  
Zheng W. et al., 2012, *Nature*, 489, 406  
Zitrin A. et al., 2009, *MNRAS*, 396, 1985  
Zitrin A. et al., 2011, *ApJ*, 742, 117

## APPENDIX A: INTERPOLATION WITH RADIAL BASIS FUNCTIONS

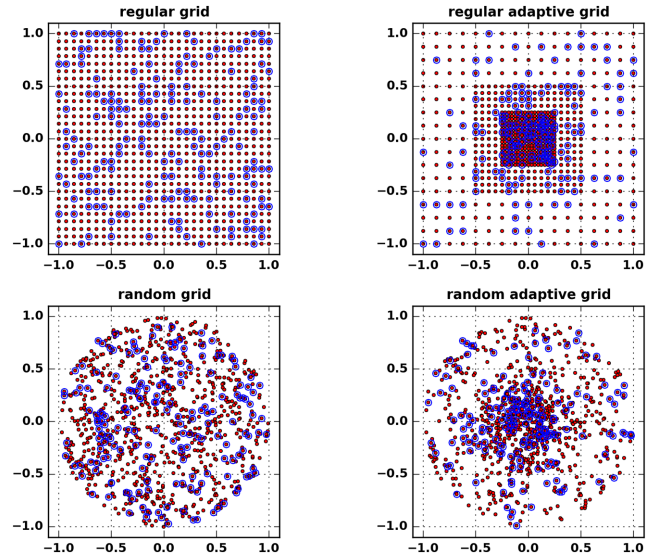
In the following we test the robustness and accuracy of the interpolation in 2D using a Gaussian radial basis function (RBF). The main free parameters in this analysis are the shape parameter of the Gaussian RBF, the number of nearest neighbours used in the interpolation stencil and the number of evaluation points to perform the interpolation.

We define the 2D lensing potential of an NFW halo (e.g. Bartelmann et al. 1996; Meneghetti et al. 2000; Wright & Brainerd 2000)  $\psi(x, y) = 4\kappa_s a(x, y)$  as a test function, with

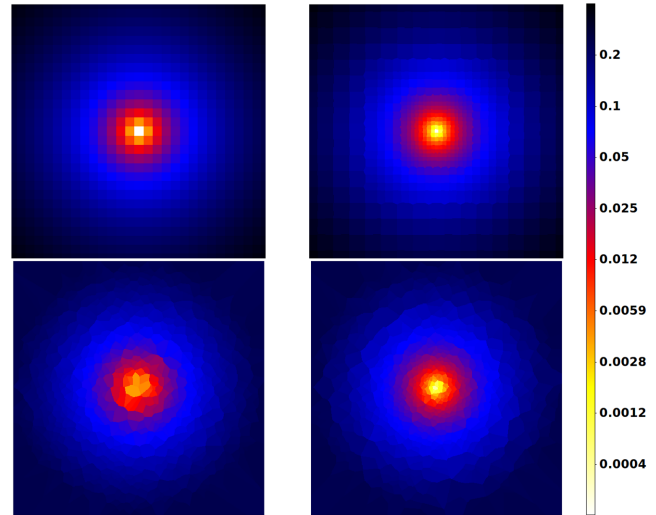
$$a(x, y) = \frac{1}{2} \ln^2 \frac{r}{2} + \begin{cases} 2 \arctan^2 \sqrt{\frac{r-1}{r+1}} & \text{for } (r > 1) \\ -2 \arctan^2 \sqrt{\frac{1-r}{1+r}} & \text{for } (r < 1) \\ 0 & \text{for } (r = 1) \end{cases} \quad (\text{A1})$$

and  $r = \sqrt{x^2 + y^2}$ . For the scale convergence we choose  $\kappa_s = 0.25$  and define four different kind of node layouts, all of which are shown in the four panels of Fig. A1. The first layout is a regular, square mesh with 900 nodes. The second mesh has the same number of total nodes but is refined twice towards the centre of the mesh. In each refinement step, the separation of neighbouring mesh nodes is reduced by a factor of 2. The third layout is defined by 900 random nodes on the unit disc, as is the fourth layout, with the difference that also this configuration is refined twice towards the centre of the disc. The last two examples define mesh-free sets of nodes. In the following, we interpolate the test function on these domains. For illustration purposes we evaluated equation (A1) on the nodes of all four configurations in Fig. A2.

In order to perform interpolations, we define as a first step a set of 180 evaluation points in each of the four test cases. The evaluation points for each node layout are shown as blue circles in



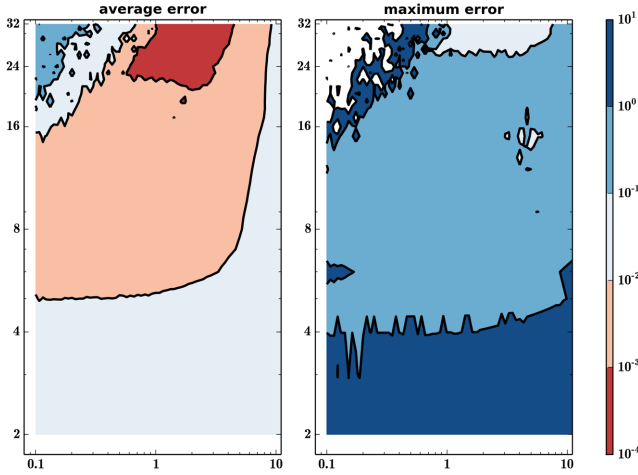
**Figure A1.** The evaluation point configurations we use for our performance tests of RBF interpolation and differentiation. The red points in the top left panel show a regular mesh with 900 nodes and no refinement. The top right panel shows a regular mesh of 900 nodes with two refinement levels towards the centre. The bottom panels show 900 randomly chosen nodes on the unit disc. The example in the bottom left panel is not refined, while the bottom right panel includes two levels of refinement towards the centre of the disc. In each refinement step, the density of random points doubles. The blue circles show an ensemble of 180 evaluation points for each setup, which anchor the interpolant in the interpolation test.



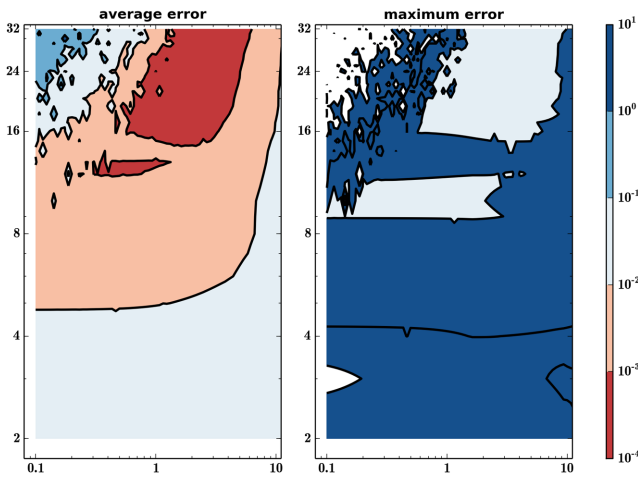
**Figure A2.** The test function defined in equation (A1) evaluated at the nodes of the four setups shown in Fig. A1.

Fig. A1. We calculate the interpolant  $\tilde{f}(x, y)$  of the test function  $f(x, y)$  by using equation (2) and define two performance metrics. The average relative error  $\langle (\tilde{f}(x, y) - f(x, y)) / f(x, y) \rangle$  for all nodes  $\mathbf{x} = (x, y)$  and the maximum relative error  $\max((\tilde{f}(x, y) - f(x, y)) / f(x, y))$  for all nodes  $\mathbf{x} = (x, y)$ . We evaluate both metrics as a function of the shape parameter and the number of nearest neighbours used to derive the interpolant at a given node  $\mathbf{x}$ . For the regular, non-refined mesh we plot the results in Fig. A3 and for the non-refined, mesh-free setup in Fig. A4. As one can see, the overall performance is excellent once the right shape parameter is found.





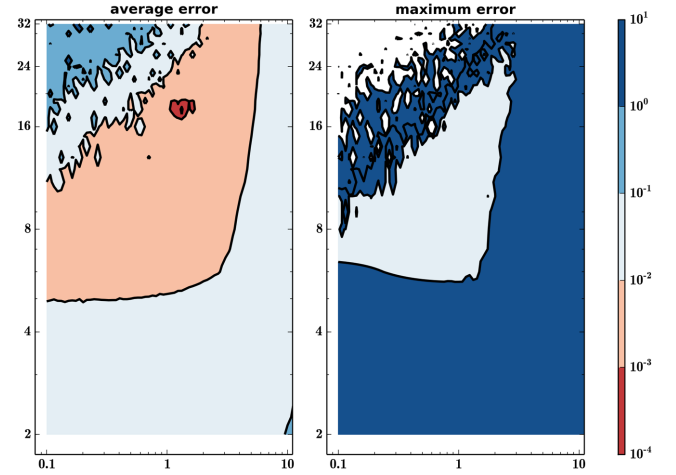
**Figure A3.** Interpolation performance on a regular, unrefined mesh with 180 evaluation points. The left-hand panel shows the average relative interpolation error as a function of the shape parameter of the Gaussian RBF and the number of nearest neighbours used to calculate the interpolant in each grid point. The right-hand panel shows the maximum relative error of all nodes.



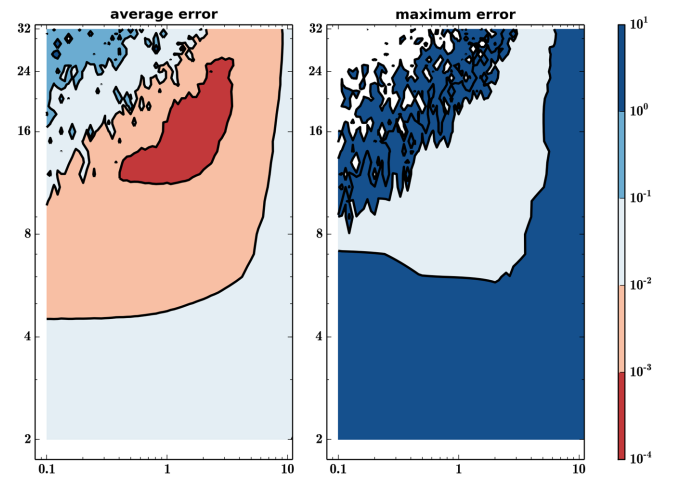
**Figure A4.** This figure shows the same plot as in Fig. A3, but for a mesh-free, unrefined node layout with 180 evaluation points.

With the use of 16 nearest neighbours or more, the average relative errors approach the  $10^{-4}$  level and the maximum error approaches  $10^{-2}$ . The performance is slightly better for the mesh-free setup, which is due to the fact that the RBF approach is not well suited to treat the edges of the regular mesh. The same holds for the interpolation on the refined node layouts. We show the results for the regular, refined mesh in Fig. A5 and for the random, mesh-free node layout in Fig. A6. The performance in the latter case is similar to the unrefined one, but the performance drops slightly for the regular, refined mesh. Also here the RBF approach is not ideal to treat the abrupt transitions between the different refinement levels, which are not well-described by a radially dependent function. However, the overall performance in all four cases is remarkable.

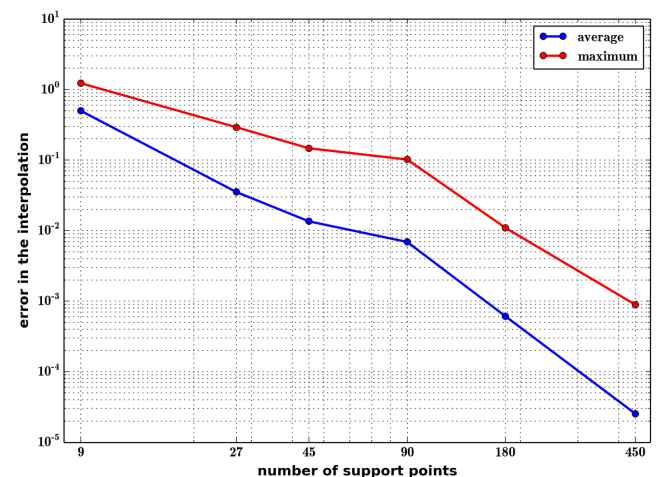
As a last test we vary the number of evaluation points in the mesh-free, refined node layout, going from 1 per cent of the total number of nodes, 9, to 50 per cent of the total number of nodes, 450. We plot both performance metrics as a function of the number of evaluation points in Fig. A7, where the shape parameters and the



**Figure A5.** This figure shows the same plot as in Fig. A3, but for a refined regular mesh with 180 evaluation points.

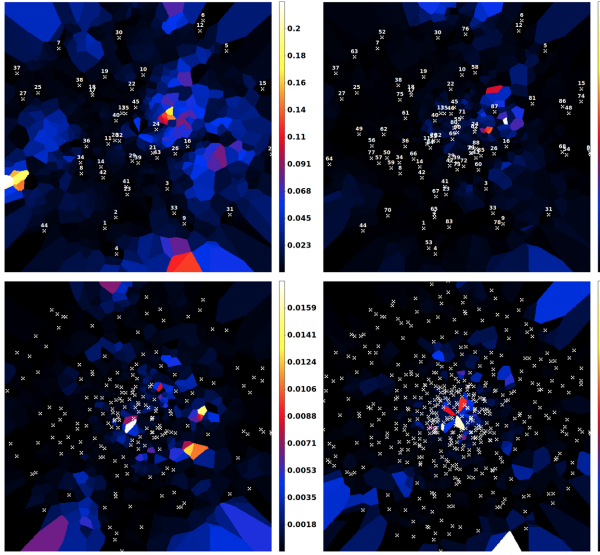


**Figure A6.** This figure shows the same plot as in Fig. A3, but for a random, mesh-free node layout with two levels of refinement towards the centre.



**Figure A7.** The accuracy of the interpolation on the mesh-free, refined node layout as a function of the number of evaluation points. The shape parameter and the number of nearest neighbours were chose to be optimal, according to the previous analysis.





**Figure A8.** A visualization of the interpolation performance for 45, 90, 180 and 450 evaluation points, respectively. The colour coding shows the relative interpolation error at each points of the mesh-free, refined node layout. The white markers in each panel show the position of the evaluation points.

number of nearest neighbours in each case are optimally chosen. The interpolation accuracy is clearly a steep function of the number of evaluation points, where very good results ( $<10^{-4}$  average and  $<10^{-2}$  maximum relative error) can be achieved with a large number of evaluation points exceeding one quarter of the total number of nodes. We investigate the dependence of the interpolation accuracy on the number and position of the support points a little further in Fig. A8, where we show the relative interpolation error for four different evaluation point setups.

## APPENDIX B: FINITE DIFFERENCING WITH RADIAL BASIS FUNCTIONS

To test the accuracy of numerical differentiation using RBF FD stencils, we return to equation (A1) and calculate two important derivatives.

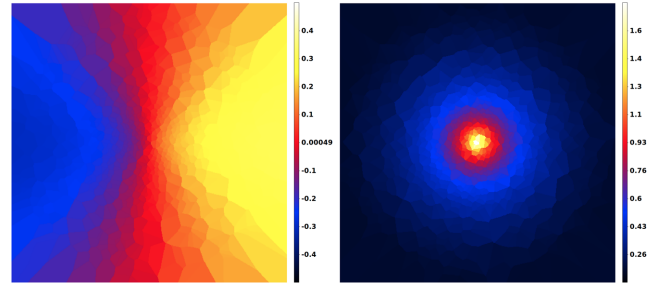
$$\frac{\partial \psi(x, y)}{\partial x} = \alpha_1(x, y) = x \frac{\kappa_s}{r^2} b(x, y) \quad (\text{B1})$$

with

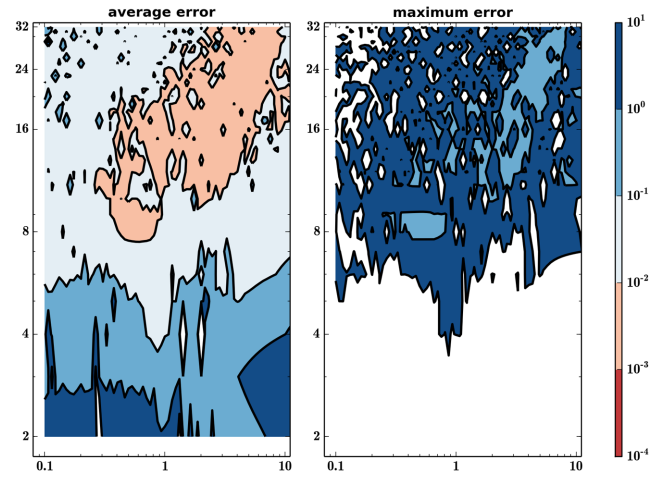
$$b(x, y) = \frac{1}{2} \ln \frac{x}{2} + \begin{cases} \frac{2}{\sqrt{r^2-1}} \arctan \sqrt{\frac{x-1}{x+1}} & \text{for } (x > 1) \\ \frac{2}{\sqrt{1-r^2}} \operatorname{arctanh} \sqrt{\frac{1-x}{1+x}} & \text{for } (x < 1) \\ 1 & \text{for } (x = 1) \end{cases} \quad (\text{B2})$$

which is the first component of the deflection angle, and

$$\frac{1}{2} \Delta \psi(x, y) = \kappa(x, y) = 2 \frac{\kappa_s}{r^2 - 1} c(x, y) \quad (\text{B3})$$



**Figure B1.** The derivatives of the test function on the mesh-free, refined node layout. The left-hand panel shows the first derivative in the  $x$  direction and the right-hand panel shows the 2D Laplacian of the function multiplied by  $1/2$ .



**Figure B2.** The accuracy of the numerical derivatives calculated by RBF FD. Shown is the average and maximum relative error on the mesh-free, refined node layout for the first  $x$  derivative of the test function.

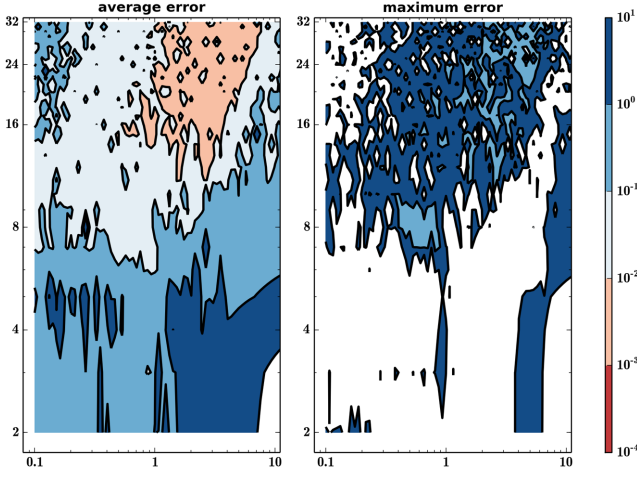
with

$$c(x, y) = \begin{cases} 1 - \frac{2}{\sqrt{r^2-1}} \arctan \sqrt{\frac{x-1}{x+1}} & \text{for } (x > 1) \\ 1 - \frac{2}{\sqrt{1-r^2}} \operatorname{arctanh} \sqrt{\frac{1-x}{1+x}} & \text{for } (x < 1) \\ \frac{1}{3} & \text{for } (x = 1) \end{cases} \quad (\text{B4})$$

which is the convergence of the NFW test potential.

In the following we perform tests on the mesh-free refined node layout of Appendix A only, since it resembles best the conditions in our real lensing applications. We visualize the test function and the three derivatives of interest for this node configuration in Fig. B1.

We use equation (4) to calculate the derivatives of the test function. This operation has the Gaussian RBF shape parameter and the number of nearest neighbours as free parameters. We again define the average and the maximum error for all nodes as performance metrics and plot them for the first  $x$  derivative in Fig. B2. As expected, the right choice of the shape parameter is crucial and the accuracy steadily increases when using more nearest neighbours to calculate the numerical derivatives. This also applies to the accuracy of the numerical Laplacian as shown in Fig. B3. With the right choice of shape parameter and an adequate number of nearest neighbours, average relative errors on the grid of  $<10^{-3}$  and maximum errors of  $<10^{-1}$  can be achieved throughout.



**Figure B3.** This figure is identical to Fig. B2 but shows the RBF FD performance for the 2D Laplacian of the test function multiplied by 1/2.

### APPENDIX C: LINEARIZATION OF THE LIKELIHOOD FUNCTION

In the methodology outlined in Section 3.2, we need to minimize a complicated  $\chi^2$ -function with the lensing potential in each evaluation point as a free parameter. This function consists of a contribution from a reduced-shear term, a critical-line estimator term, a multiple-image system term and a regularization term. The solution to this system is the wanted lensing potential. Many of the explicit calculations have already been carried out in Bradač et al. (2005a) and Merten et al. (2009) which is why we only carry out the derivation of the multiple-image system term, not part of Merten et al. (2009) and where we use a slightly different approach than Bradač et al. (2005a).

Starting point is equation (26) which we partially differentiate after  $\psi_k$  and use equations (18) and (22).

$$\begin{aligned} \frac{\partial \chi_m^2}{\partial \psi_l} = & \sum_{n=1}^{N_s} \frac{2}{\sigma_s^2} \left[ \theta_n \mathcal{D}_{nl} + \mathcal{D}_{nk} \mathcal{D}_{ni} \psi_k \right. \\ & + \frac{1}{N_s} \sum_{i=1}^{N_s} (\theta_n \mathcal{D}_{il} - \mathcal{D}_{il} \mathcal{D}_{nk} \psi_k) \\ & + \frac{1}{N_s} \sum_{i=1}^{N_s} (\theta_i \mathcal{D}_{nl} - \mathcal{D}_{nl} \mathcal{D}_{ik} \psi_k) \\ & \left. - \frac{1}{N_s^2} \sum_{i,j=1}^{N_s} (\theta_j \mathcal{D}_{il} - \mathcal{D}_{jk} \mathcal{D}_{il} \psi_k) \right] \end{aligned} \quad (C1)$$

after sorting terms with and without  $\psi_k$ -term, we find the linear system of equations

$$\begin{aligned} \mathcal{B}_{lk}^m = & \sum_{n=1}^{N_s} \frac{2}{\sigma_s^2} \left[ \mathcal{D}_{nl} \mathcal{D}_{nk} - \frac{1}{N_s} \sum_{i=1}^{N_s} (\mathcal{D}_{nk} \mathcal{D}_{il} + \mathcal{D}_{nl} \mathcal{D}_{ik}) \right. \\ & \left. + \frac{1}{N_s^2} \sum_{i,j=1}^{N_s} \mathcal{D}_{il} \mathcal{D}_{jk} \right] \end{aligned} \quad (C2)$$

and data vector

$$\begin{aligned} \mathcal{V}_l^m = & \sum_{n=1}^{N_s} \frac{2}{\sigma_s^2} \left[ \theta_n \mathcal{D}_{nl} - \frac{1}{N_s} \sum_{i=1}^{N_s} (\theta_n \mathcal{D}_{il} + \theta_i \mathcal{D}_{nl}) \right. \\ & \left. + \frac{1}{N_s^2} \sum_{i,j=1}^{N_s} \theta_j \mathcal{D}_{il} \right] \end{aligned} \quad (C3)$$

where  $\theta$  is one component of the observed lens-plane position of the multiple images of the system. In order to obtain the full multiple image term contribution, one has to sum the contributions from both coordinate components by substituting  $\theta$  with  $\theta_1$  or  $\theta_2$  and  $\mathcal{D}$  with  $\mathcal{D}^1$  or  $\mathcal{D}^2$ , respectively. If there are more than one multiple-image systems in the lens, each system contributes a  $\chi^2$ -term.

In order to obtain the full linear system of equations, all individual contributions to an entry in the coefficient matrix  $\mathcal{B}_{lk}$  or in the data vector  $\mathcal{V}_l$  need to be summed

$$\mathcal{B}_{lk} = \mathcal{B}_{lk}^w + \mathcal{B}_{lk}^s + \mathcal{B}_{lk}^m + \mathcal{B}_{lk}^{s,\text{reg}} + \mathcal{B}_{lk}^{c,\text{reg}} \quad (C4)$$

$$\mathcal{V}_l = \mathcal{V}_l^w + \mathcal{V}_l^s + \mathcal{V}_l^m + \mathcal{V}_l^{s,\text{reg}} + \mathcal{V}_l^{c,\text{reg}}. \quad (C5)$$

This paper has been typeset from a  $\text{\LaTeX}$  file prepared by the author.